**УЧЕБНЫЕ ЗАДАНИЯ**

**ПО РАБОТЕ С ПРОГРАММНЫМ ОБЕСПЕЧЕНИЕМ**

**«Среда динамического моделирования технических систем «SimInTech»**

на примере создания простейшей комплексной модели

ООО «ЗВ Сервис», 2016

# Введение. Концепция создания комплексной модели объекта в среде SimInTech

Среда динамического моделирования технических систем «SimInTech» является гибким и мощным средством для разработки сложных (другое название – комплексных) расчетных моделей. Создание таких моделей требует особого подхода и использования специального инструмента для подготовки модели, при сохранении возможности ее разделения на более простые части для удобств коллективной работы, и создания из них единой комплексной модели. В комплексной модели могут быть объединены расчеты различных физических процессов, в том числе рассчитываемые различными расчетными кодами (теплогидравлическими, электрическими и др.).

Одной из основных особенностей использования SimInTech для комплексных моделей является идеология использования «**Базы данных сигналов**» - структурированного списка переменных, обеспечивающих обмен расчетными значениями между расчетными схемами в единой модели.

**База данных сигналов** SimInTech является объектной и обеспечивает пользователю удобное решение следующих задач:

* Объединение нескольких расчетных схем в единую модель;
* Обеспечение возможности векторной обработки сигналов для типовых алгоритмов управления;
* Объектно-ориентированное проектирование модели технических систем;
* Автоматизацию создания и обработки переменных в комплексных моделях.

Общий вид концепции применения базы данных сигналов для комплексного моделирования представлен на рисунке:

Рисунок В.1. Концепция базы данных сигналов и комплексной модели в SimInTech

При создании модели с использованием базы данных сигналов возможно разделение сложной комплексной модели на произвольное количество более простых расчетных схем и отладка каждой схемы, сначала автономно, потом в составе комплексной модели.

Использование базы данных сигналов позволяет подключать сторонние расчетные коды для моделирования специализированных физических процессов. База данных сигналов обеспечивает также подключение реальной аппаратуры управления к модели, для тестирования аппаратуры управления на математической модели объекта.

В следующих 10 упражнениях на простейшей модели будет показан пример создания комплексной модели с использованием механизма базы данных сигналов.

Для создания простых моделей возможности базы данных сигналов могут показаться лишними и обременительными, однако для комплексных моделей (например, комплексная математическая модель динамики ЯЭУ), использование предложенного подхода дает огромное преимущество перед конкурентными моделирующими программными продуктами.

## Первый запуск среды SimInTech

Перед тем, как приступать к выполнению вводного курса, состоящего из 10 пошаговых упражнений, рассмотрим саму среду динамического моделирования SimInTech. Порядок установки и регистрации описан в «Инструкции по установке и регистрации ПО SimInTech» (доступна для скачивания на сайте www.simintech.ru).

Запуск среды SimInTech осуществляется одним из двух способов:

1. С помощью ярлыка, появляющегося на рабочем столе после установки SimInTech;
2. В зависимости от версии операционной системы (применительно к ОС Windows):

* Windows XP, 7, 10 – с помощью стартового меню «**Пуск**»;
* Windows 8, 8.1 – с помощью выбора соответствующего ярлыка на экране приложений.

После запуска приложения на экране появится Главное Окно среды динамического моделирования SimInTech (рисунок В.2)

Рисунок В.2. Главное Окно SimInTech

Данное окно можно поделить на три области:

* 1 область – Главное Меню;
* 2 область – Панель инструментов (кнопок);
* 3 область – Палитра блоков.

Рассмотрим за что отвечает каждая из областей подробнее.

**Главное меню** позволяет производить настройку программы под конкретные нужды пользователя, управлять созданием, редактированием и расчетом сформированных схем, содержит инструменты работы с базой данных, настройки визуального вида схемного окна и пр.

**Панель инструментов** является настраиваемой областью с набором кнопок, отвечающих за некоторые часто используемые команды. Данная панель служит для ускорения доступа к этим командам. Кнопки, содержащиеся в панели, разбиты по группам в соответствии с выполняемыми функциями.

Отображение групп кнопок пользователь может изменять по своему усмотрению. Для того, чтобы отобразить или скрыть нужные кнопки, необходимо кликом правой кнопки мыши по области меню вызвать выпадающее меню, в котором затем нужно проставить или снять галочки напротив нужных пользователю групп кнопок (рисунок В.3).

Рисунок В.3. Контекстное меню настройки панелей инструментов в Главном Окне SimInTech

**Палитра блоков** – это набор библиотек блоков, который используется для формирования расчетных схем. Создание расчетных схем с помощью блоков – один из способов разработки расчетных моделей. Стоит отметить, что набор блоков в Палитре блоков изменяется в зависимости от выбранного типа решателя схемы, т.е. когда мы выбираем решатель «Автоматика», то отображается палитра блоков для реализации схем автоматики, а когда мы выбираем решатель «Теплогидравлика», то отображается уже набор блоков для реализации теплогидравлических моделей.

Создать новую расчетную схему можно двумя способами:

1. Выбрать в Главном меню пункт «**Файл**», далее пункт «**Новый проект**» и из выпадающего списка выбрать нужный тип расчетной схемы;
2. Воспользоваться кнопкой «**Новый проект**» в панели инструментов (данная кнопка находится в группе «**Файл**», поэтому если она не отображена в основном окне, то необходимо вызвать меню активации групп и активировать для отображения группу «**Файл**»). Затем выбрать нужный тип файла. В зависимости от установленной конфигурации среды SimInTech может отображаться разный набор типов файлов. В базовой версии среды SimInTech доступны следующие типы:

* Схема TPP;
* Схема автоматики;
* Схема теплогидравлики;
* Пакет;
* Пустой проект.

После выполнения одного из вышеупомянутых способов на экране появится новое окно. Для данного окна полностью применим функционал стандартных методов операционной системы при работе с окнами: можно изменять визуальные размеры окна, его расположение на экране, можно сворачивать окно, разворачивать окно на весь экран и пр. При сохранении проекта сохраняются и визуальные настройки окна, такие как размеры и расположение на экране.

При открытии окна, когда уже ранее открыто несколько окон, данное окно будет активным для редактирования, остальные же окна автоматически перейдут в пассивный режим. Активировать их можно одиночным кликом левой кнопки мыши по окну либо вызвав данное окно из строки состояния.

Рассмотрим подробнее области данного окна:

* 1 область – меню окна проекта;
* 2 область – панель инструменты настройки параметров расчета схемы, управления расчетом;
* 3 область – форма расчетной схемы;
* 4 область – строка отображения расчетной информации.

**Меню окна проекта** отвечает за основные функции работы со схемой, такие как создание новой схемы, сохранение схемы, открытие ранее созданных схем, вызов базы данных, предоставление доступа к справочным материалам, а также настройку визуального отображения схемы.

**Панель инструментов** настройки параметров расчета схемы содержит две группы объектов для настройки и управления расчетной схемой. Данные группы можно отобразить или скрыть на схемном окне. Для этого можно либо пройти в меню «**Вид**» - «**Панель инструментов**» и активировать/деактивировать соответствующие пункты, либо необходимо кликом правой кнопки мыши по области вызвать выпадающее меню, в котором можно сделать то же самое.

Рассмотрим подробнее содержание данных групп.

Рисунок В.4. Панели инструментов Схемного Окна SimInTech

В первой группе содержатся следующие объекты (рисунок В.4):

* кнопка доступа к окну изменения расчетных параметров схемы;
* поле активации и отображения выбранного пользователем решателя;
* кнопка выбора режима отображения схемы;
* кнопка работы со слоями;
* кнопка активации показа значений на линиях связи.

Во второй группе содержатся кнопки управления расчетом (рисунок В.4):

* **инициализация** – инициализирует расчетную схему, но не запускает ее расчет;
* **пуск** – инициализирует расчетную схему и сразу запускает ее расчет;
* **сделать шаг** – задача делает один шаг с величиной, заданной пользователем в расчетных параметрах схемы;
* **пауза** – приостанавливает расчет;
* **стоп** – останавливает расчет.

Форма расчетной схемы имеет два режима отображения (рисунок В.4): режим схемы и режим скрипта.

В **режиме схемы** на форму можно помещать блоки, находящиеся в палитре блоков, с помощью них создавать расчетные схемы, редактировать ранее набранные схемы, создавать собственные блоки и многое другое.

В **режиме скрипта** активен редактор программного кода встроенного Языка программирования, в котором можно решить с помощью встроенных операторов и функций, например, такие задачи, как:

* расчет математических выражений и уравнений;
* управление отображением графических примитивов;
* реализация простых алгоритмов управления для настройки и отладки теплогидравлических моделей.

Это лишь небольшой перечень задач, с которыми в состоянии справиться встроенный язык программирования, возможности которого гораздо шире.

При открытии новой, либо ранее сохраненной схемы всегда активирован режим схемы. Активация режима схемы или режима скрипта осуществляется с помощью одиночного клика левой кнопкой мыши по соответствующей закладке с именем режима.

Выбрать нужный блок и поместить его на форму просто:

1. Нужно выбрать необходимую вкладку в Палитре.
2. Выбрать одиночным нажатием тот блок, который необходимо установить на форме. При этом выбранный блок будет подсвечен в Палитре блоков.
3. При появлении курсора в пределах формы расчетной схемы, вид курсора изменится: появится выбранный блок, а вместо курсора появится крестообразный указатель места установки для этого блока. Выбрать место установки блока.
4. Одиночным нажатием левой кнопки мыши на форме установить блок.

Задание свойств блока осуществляется при помощи окна задания свойств. Вызов этого окна можно осуществить двумя способами:

1. Выделив блок одиночным кликом правой кнопки мыши вызвать выпадающее меню, в котором выбрать пункт «Свойства объекта»;
2. Выделив блок нажать на панели кнопок кнопку «Свойства».

После этого на экране появится окно (рисунок В.5).

Рисунок В.5. Окно свойств блока, на примере блока типа «Синусоидальный сигнал».

В окне свойств в зависимости от типа блока могут отображаться следующие вкладки:

* **Свойства** – расчетные свойства блока, непосредственно влияющие на его исполнение;
* **Общие** – базовые свойства блока, такие как имя, тип, координаты расположения на листе, видимость и другие;
* **Порты** – список и редактор параметров входных/выходных портов блока;
* **Визуальные слои** – редактор расположения блока на одном или нескольких визуальных слоях.

Наиболее часто используемыми вкладками данного окна являются «Свойства» и «Общие».

Примечание: при разработке проектов, особенно скриптов, бывает необходимость обращаться к свойствам и параметрам блоков по именам свойств. Для этого бывает удобно отобразить колонку «Имя» в редакторе свойств блоков, скрытую по умолчанию. Это можно сделать через пункт меню «**Вид → Показать поле «Имя» в редакторе свойств блока**».

# Создание схемы автоматики с базой сигналов задвижек и датчиков

## Создание новой схемы автоматики

Для создания схемы автоматики в SimInTech надо выполнить следующие действия:

1. В главной панели инструментов выбрать кнопку «Новый проект»;
2. В выпадающем меню выбрать пункт «Схема автоматики» (Рисунок 1, второй вариант справа).

Рисунок 1. Меню создания нового проекта (два варианта)

После этого откроется новое схемное окно, в котором и будет происходить создание структурной схемы системы автоматики (Рисунок 2).

**Примечание**: выбор вариантов новых проектов зависит от комплектации установленного ПО SimInTech. Пользователь может разработать свои «шаблоны» новых проектов, и при размещении в директории C:\SimInTech\bin\Template они станут доступны к выбору через Главное меню.

Рисунок 2. Схемное окно для создания схемы автоматики.

Для дальнейшей работы необходимо сохранить данную схему в файле с новым именем (для примера, «**Схема автоматики 1.prt**»). Чтобы это сделать, нужно произвести следующие действия:

1. В главном меню выбрать пункт «Файл → Сохранить проект как...».
2. Используя стандартный диалог сохранения файла, выбрать новый каталог и имя файла. В данном и последующих упражнениях сохраним файл как: «C:\SimInTech\Projects\Схема автоматики 1.prt».

## Подключение базы данных сигналов

По умолчанию расчетные схемы математических моделей предназначены для автономного локального моделирования, поэтому подключение к какому-либо файлу базы данных отсутствует (есть возможность задавать переменные проекта через скрипт либо через плоский список сигналов в пункте Главного меню «Сервис → Сигналы...»). Однако к любой схеме в SimInTech можно подключить базу данных сигналов для структурного хранения глобальных констант и переменных, а также для организации взаимодействия как между расчетными схемами, так и со сторонними расчетными программами. Общая для нескольких проектов база сигналов позволяет решить большое количество различных задач, например, таких:

* организовать передачу значений сигналов между различными решателями и расчетными схемами;
* организовать хранение общих констант для нескольких расчетных схем, с возможностью оперативного доступа к их просмотру и изменению в процессе отладки модели;
* отслеживать изменение значений сигналов в процессе расчета;
* реализовать векторную обработку сигналов.

**Примечание**: для дальнейшей работы требуется перевести среду SimInTech в режим разработчика. Активировать данный режим можно двумя способами:

1. В меню основного окна выбрать пункт «**Вид** → **Режим разработчика**» (Рисунок 3).

Рисунок 3. Включение режима разработчика из меню основного окна

1. В меню основного окна выбрать пункт «**Файл → Параметры…**», после чего появится диалоговое окно, где во вкладке «**Вид**» следует активировать первый пункт «**Режим разработчика**». Через параметры программы настраиваются те опции, которые влияют на поведение установленного экземпляра ПО SimInTech в целом, для всех проектов.

Подключение базы данных сигналов к схеме автоматики осуществляется следующим образом:

1. На схемном окне нажать кнопку «Параметры расчета»:

Рисунок 4. Кнопка доступа к параметрам расчета

1. В появившемся диалоговом окне настроек перейти на закладку «Настройки». (Рисунок 5). Данная закладка является невидимой если не включён режим разработчика.
2. В поле ввода «Модуль базы данных проекта» необходимо ввести следующий текст: $(Root)\sdb.dll, где sdb.dll – имя динамической библиотеки программного модуля базы данных сигналов, поставляемой в составе стандартного дистрибутива; $(Root) – служебное имя, указывающее по умолчанию на папку C:\SimInTech\bin (чему именно равно $(Root), настраивается в параметрах программы, но как правило значение по умолчанию не изменяется в большинстве случаев применения ПО). В поле ввода «Модуль базы данных проекта» можно ввести и C:\SimInTech\bin\sdb.dll, что равнозначно введённому $(Root)\sdb.dll.
3. В поле ввода «Имя базы данных проекта» ввести произвольное имя файла для сохранения базы данных. В рамках данного и последующих упражнений присвоим имя базе сигналов signals.db, однако имя базы сигналов в общем случае может быть любым.

В случае, когда не прописывается полный путь к файлу, по умолчанию он располагается в том же каталоге, что и файл проекта). Можно использовать относительные пути (например, «**..\База\signals.db**»).

Рисунок 5. Закладка настройки базы данных проекта

1. Закрыть диалоговое окно нажатием кнопки «Ок» (Рисунок 5).
2. Выполнить сохранение текущего проекта, нажав кнопку «Сохранить проект» на главном окне программы (Рисунок 6).

Рисунок 6. Кнопка сохранения текущего проекта

При сохранении проекта с подключенной базой сигналов, файл базы сигналов должен появиться на диске по указанному пути и с указанным именем. Это можно проверить, перейдя (проводником или другим файловым менеджером) в каталог проекта. В случае, если проект необходимо подключить к уже существующей на диске базе сигналов, после привязки проекта к ней и при сохранении проекта следует ответить **Нет** на запрос пересохранения базы. Иначе существующая база сигналов будет перезаписана пустым файлом. База сигналов считывается из файла только при загрузке проекта, а не при прописывании файла базы сигналов в проекте.

## Интерфейс просмотра и редактирования базы данных сигналов

Работа по наполнению и редактированию базы данных сигналов осуществляется с помощью модуля «Редактор базы данных». Вызов базы данных сигналов осуществляется через главное меню программы (пункт меню основного окна «Инструменты», подпункт «База данных...», Рисунок 7).

Рисунок 7. Вызов редактора базы данных

Внимание!!! Для проектов, в которых не используется база данных, данный пункт меню является недоступным. Чтобы получить доступ к этому пункту меню, необходимо выполнить последовательность действий, описанную в разделе 1.2.

После выбора пункта меню появляется диалоговое окно редактора базы данных проекта (Рисунок 9).

Редактор базы данных содержит три закладки, расположенные в верхней части окна: непосредственно «Редактор», «Настройки», «**Состояние сети**» (Рисунок 8). Для работы с сигналами используется закладка «Редактор».

Рисунок 8. Вкладки окна базы данных сигналов.

Окно редактора разбито на несколько областей. Три основные области (панели): «Категории», «Группы сигналов» и «Сигналы и данные для групп». Также доступна вспомогательная панель **«Фильтры»**, в которой можно создавать и сохранять пользовательские фильтры для отображения только части сигналов базы данных, отфильтрованных по какому-либо критерию. Этот механизм используется для навигации и удобной работы с большими базами данных.

Панель «Категории» содержит список категорий объектов, объединенных по каким-либо общим признаками, которые могут быть помещены в базу данных. К примеру, «БУЗ» - категория, объединяющая блоки управления задвижками, **«БУД»** - категория, объединяющая блоки управления двигателями, **«Общие»** - категория, объединяющая сигналы, использующиеся в проекте, но не имеющие каких-либо явных признаков, по которым можно отнести их к другим категориям (что и является общим признаком для отнесения их к данной категории).

Панель Группы сигналов содержит список конкретных объектов (экземпляров), имеющихся в базе данных проекта, которые соответствуют выбранной категории. Например, 1L01AS1\_2 – код и имя конкретной задвижки, существующей как в базе данных, так и в расчетной модели объекта. Как правило, сигналы имеющиеся в группе сигналов, соответствуют шаблону категории.

Панель Сигналы и данные для групп содержит сводную таблицу с сигналами и данными, соответствующими конкретной группе сигналов, выбранной в панели Группы сигналов.

Пользователь может самостоятельно создавать, редактировать и удалять новые категории объектов, шаблонные сигналы каждой категории, перечень объектов (групп сигналов) в каждом проекте, а также список и значения сигналов.

Рисунок 9. Диалоговое окно «Редактор базы данных проекта»

По умолчанию для вновь создаваемого проекта, все панели редактора не содержат элементов, база данных сигналов пуста. Пользователь может как самостоятельно наполнить базу данных, так и осуществить считывание ранее подготовленного файла существующей базы данных. Либо произвести импорт из текстового или некоторых других форматов (csv, xml, sql), подготовленных специальным образом.

Для редактирования списков категорий, групп сигналов, фильтров, а также сигналов, содержащихся в сводной таблице, под каждой из панелей присутствуют кнопки «**Добавить**» и «**Удалить**» .

## Работа с редактором базы данных

Добавление новой категории осуществляется следующим образом:

1. Пользователь должен нажать кнопку «Добавить категорию» в нижней части панели **Категории**. После этого в списке категорий появляется запись «Новая категория».
2. Двойное нажатие на элементе «Новая категория» открывает диалоговое окно редактирования шаблона категории и шаблонных сигналов «**Шаблон категории**» (Рисунок 10). В окне присутствуют следующие элементы:

* **таблица шаблонных сигналов**, которые соответствуют данной категории. По сути это те сигналы, которые будут автоматически присутствовать в группе сигналов, которая создается в рамках данной категории. Например, для задвижки это могут быть вещественное значение положения штока, двоичное значение сигнала «открыта» или «закрыта» и т. д.; сигналы, задаваемые здесь, не являются еще сигналами проекта, это только заготовка для добавляемых позднее групп сигналов в этой категории;
* **поле ввода имени категории**. Посредством данного поля мы задаем имя категории. По умолчанию новая добавляемая категория имеет имя «Новая категория»;
* **поле ввода шаблона имени групп**. В этом поле вводится та часть имени группы, которая автоматически появляется в окне добавления новой группы;
* **кнопки редактирования таблицы сигналов**, которые позволяют: добавить сигнал, удалить сигнал, очистить список сигналов, загрузить список сигналов из файла, сохранить список в файл, копировать сигнал в буфер памяти, вставить сигнал из буфера памяти, переместить сигнал в таблице выше или ниже.

Рисунок 10. Диалоговое окно редактирования свойства категории

Представленное на рисунке диалоговое окно содержит простейший образец заполнения свойств категории типа «Задвижки». Для дальнейшей работы нам понадобятся данные сигналы, поэтому необходимо заполнить таблицу, подобно тому, как изображено на рисунке (Рисунок 10), и сохранить сделанные изменения.

Для сохранения изменений необходимо закрыть диалоговое окно «Свойства категории» нажатием на кнопку «Ок». После этого в списке категорий появится новая категория с именем «Задвижки».

Рисунок 11. Редактор базы данных с добавленной категорией (выделена кнопка «Добавить таблицу»)

Для каждой созданной категории пользователь может добавить в базу данных неограниченное количество групп сигналов. Например, создав категорию задвижек определенного типа, можно добавить в базу данных все задвижки данного типа, используемые в расчетной модели. Для этого необходимо:

1. Выделить название категории в панели категорий (в нашем примере это категория «Задвижки»).
2. Нажать кнопку «Добавить таблицу» в панели групп сигналов (Рисунок 11).
3. В диалоговом окне «Создание новых групп» вести имена групп сигналов данной категории, которые необходимо внести в базу данных: (Рисунок 12).
4. Закрыть диалоговое окно «Создание новых групп» нажатием кнопки «**Ок**».

Рисунок 12. Диалоговое окно создания новых групп сигналов (например, для задвижек)

Имена групп сигналов желательно задавать с помощью латинских букв и цифр. Это необходимо при использовании базы данных для обработки сигналов, приходящих по протоколу ОPC, либо чтобы избежать путаницы при генерации кода. Сигналы и группы сигналов лучше начинать именовать с буквы а не с цифр.

Заданные группы сигналов появляются в панели «Группы сигналов».

Создайте аналогично тому, как была ранее создана категория «**Задвижки**», новую категорию «**Датчики**».

На рисунке (Рисунок 13) представлено диалоговое окно редактирования для категории «Датчики» в учебном примере. Заполните таблицу шаблонных сигналов для этой категории в соответствии с рисунком.

Рисунок 13. Диалоговое окно свойства категорий для категории «Датчики»

После создания новой категории, для того чтобы тестовая база данных содержала сигналы, представленные на рисунке 12, сформируйте в соответствии с описанной выше процедурой группы сигналов, принадлежащие категории «Датчики». Для иллюстрации возможностей базы данных сигналов мы создали три группы сигналов (D1, D2 и D3) для датчиков, однако далее будут использованы датчики по необходимости.

После добавления в БД категорий и групп сигналов можно осуществлять переключение между ними, выбирая нужную категорию и группу сигналов (Рисунок 14). Выбранная категория и выбранная группа сигналов, для которой отображаются в правой части интерфейса сигналы, подсвечивается синим цветом.

Остальные возможности интерфейса базы сигналов будут рассмотрены по мере необходимости, а также описаны в справочной системе ПО SimInTech.

Рисунок 14. Редактор базы данных с добавленной категорией датчики и тремя датчиками

## Сохранение базы данных проекта

Прежде чем закрывать редактор, необходимо убедиться, что настройки предполагают автоматическое сохранение базы данных. Для этого необходимо перейти на вкладку «Настройки» и установить галочки в пунктах «Сохранять базу» и «Резервировать БД» (Рисунок 15). Пункт «**Сохранять базу**» реализует алгоритм сохранения базы данных при сохранении проекта, к которому она подсоединена. Пункт «**Резервировать БД**» реализует алгоритм сохранения базы данных в файл, обеспечивая при этом хранение ранее сохраненных версий базы данных c расширениями .1 .2 .3 и так далее (пользователь при этом может в любой момент времени «откатиться» к более ранней версии базы данных, если, например, в нее были ошибочно внесены какие-либо изменения).

Рисунок 15. Настройки сохранения базы данных

Установив опции «Сохранять базу» и «Резервировать БД», необходимо переключиться на закладку «Редактор» и закрыть диалоговое окно «Редактор базы данных» нажатием кнопки «Ок». После этого необходимо сохранить проект. Данные установки позволяют автоматически сохранять базу данных проекта (файл с базой данных проекта) при закрытии файла самого проекта.

Отметим, что в файле базы сигналов (signals.db в настоящем упражнении) сохраняются только сигналы (категории с шаблонными сигналами, группы сигналов и сами сигналы с информацией). Всё остальное – настройки подключения, автосохранения и т.п. хранится в файле проекта т.к. к одной и той же базе сигналов возможно подключение разных проектов с разными настройками. Файл базы сигналов необходим только при загрузке проекта для инициализации общей области памяти с необходимым набором категорий, групп сигналов и сигналов. В отдельном файле с расширением .filters сохраняются глобальные фильтры для базы сигналов.

Если все вышеописанные действия были выполнены правильно, то при закрытии проекта возникает диалог, с помощью которого пользователь принимает решение о том, сохранять ли в существующую базу данных внесённые изменения или нет. В случае согласия пользователя с внесенными в базу данных изменениями в директории, в которой был сохранен проект, должен появиться файл с именем «signals.db» (при первичном сохранении базы данных).

Так же пользователь может принудительно сохранить базу данных в любой момент времени, не закрывая проект. Для этого необходимо нажать кнопку «**Сохранить**» на вкладке «**Настройки**». Кнопка «**Сохранить в файл**», которая находится на вкладке «Редактор», используется для сохранения базы сигналов в какой-либо файл с другим именем и/или форматом. Файл проекта при этом остается привязанным к той базе которая прописана в параметрах проекта.

# Создание файла теплогидравлической схемы с подключением базы данных сигналов

## Создание новой теплогидравлической схемы

Для создания новой теплогидравлической схемы необходимо выполнить следующие действия:

1. В главной панели инструментов выбрать кнопку «Новый проект».
2. В выпадающем меню выбрать пункт «Схема теплогидравлики» (Рисунок 16). Другие пункты относятся к созданию схемы автоматики и схемы для расчетного кода TPP. В зависимости от комплектации установки SimInTech могут быть доступны и другие расчетные схемы.

Рисунок 16. Меню создания нового проекта теплогидравлики

После это появится новое схемное окно, в котором будет происходить создание расчетной схемы теплогидравлической модели (иногда говорят, нодализационной схемы, Рисунок 17).

Рисунок 17. Схемное окно для создания схемы теплогидравлической модели

Для дальнейшей работы необходимо сохранить данную схему в файл в той же директории, в которой была сохранена ранее созданная схема автоматики. Для этого необходимо выполнить следующие шаги:

1. В главном меню выбрать пункт «Файл → Сохранить проект как...»**.**
2. Используя стандартный диалог сохранения файла, выбрать новое имя для сохранения. В рамках данного и последующих упражнений присвоим имя Схема теплогидравлики 1.prt. Каталог сохранения – тот же, что и для схемы автоматики, созданной в предыдущем задании.

## Подключение базы данных сигналов

Для совместной работы нескольких проектов, находящихся в разных файлах, необходимо, чтобы они использовали одну и ту же базу сигналов. В нашем случае база сигналов и созданные в ней сигналы для задвижек Z1 и Z2, а также сигналы для датчиков, находятся в ранее созданном файле с именем signals.db. Помните, что для подключения базы данных нужно, чтобы был активирован режим разработчика. Активация данного режима описана в пункте 1.2.

Подключение базы данных сигналов к схеме теплогидравлики осуществляется следующим образом:

1. В схемном окне теплогидравлики нажать кнопку «Параметры расчёта»:

Рисунок 18. Кнопка доступа к параметрам расчета

1. В появившемся диалоговом окне настроек перейти на закладку «Настройки» (Рисунок 19).
2. В строке редактирования «Модуль базы данных проекта» необходимо ввести следующий текст: $(Root)\sdb.dll. Текст аналогичен тексту в схеме автоматики.
3. В поле ввода редактирования «Имя базы данных проекта» ввести имя файла базы данных. Для использования ранее созданного файла базы данных нужно ввести то имя, под которым этот файл был сохранен. Ранее база данных была сохранена в файле под именем **signals.db**, поэтому необходимо ввести данное имя в поле ввода (Рисунок 19).

Рисунок 19. Закладка настройки базы данных проекта

1. Закрыть диалоговое окно нажатием кнопки «Ок».

Примечание: при подобном добавлении базы сигналов после нажатия кнопки «**Ок**» считывание базы сигналов из файла не произойдёт, необходимо закрыть и снова открыть проект чтобы он загрузился с прописанной в настройках базой сигналов. Однако прежде чем это сделать, выполним еще ряд действий.

## Настройка сохранения базы данных при ее совместной работе с несколькими проектами.

Если база данных сигналов используется с несколькими расчетными схемами, то необходима настройка сохранения базы данных в разных проектах, для того чтобы изменения в базу сигналов вносились только из какого-то одного проекта (во избежание конфликтов записи, некорректного перезаписывания файла и/или потери информации, сохраненной в него из другого проекта).

Возможны несколько вариантов реализации сохранения. Рассмотрим три основных варианта:

1. Сохранение базы данных осуществляется при сохранении любого проекта, к которому привязана база данных. При использовании данного варианта при сохранении любого файла, использующего базу данных, файл базы данных так же будет пересохранён. При этом, если пользователь работал сразу с несколькими проектами, использующими базу данных, и совершал правку базы данных, вызывая ее из разных проектов, то возможны ситуации, когда данные, внесенные в базу, не будут сохранены из-за перезаписи файла базы данных в момент сохранения проектов. Данный вариант требует от пользователя высокой внимательности при реализации и, как правило, не используется; либо используется совместно с какой-либо системой контроля за версиями файлов (Git, SVN, Mercurial и подобных).
2. Сохранение базы данных при сохранении какого-либо ограниченного числа проектов (1-2 проекта), к которым привязана база данных. При реализации этого варианта приоритет сохранения базы данных при сохранении проекта будет отдан 1-2 проектам, ее использующим. Сохранить базу данных из остальных проектов, не обладающих приоритетом, можно будет только вручную, принудительно в редакторе базы данных сигналов. Этот вариант предпочтительнее для начинающих пользователей.
3. В больших проектах с участием нескольких разработчиков автоматическое сохранение базы данных отключают у всех проектов, и база данных редактируется отдельно, вручную сохраняется, и как правило одним-двумя разработчиками. Остальные участники пользуются базой сигналов без её редактирования.

В ПО SimInTech существуют также инструменты поддержки коллективной разработки – базовая интеграция с Git/SVN, инструменты сравнения проектов между собой, инструменты сравнения и слияния баз сигналов.

Для сохранения существующей базы данных сигналов, созданной при создании схемы автоматики, будем использовать второй вариант, в котором роль приоритетного файла будет принадлежать файлу схемы автоматики, а роль неприоритетного – файлу схемы теплогидравлики.

Для этого необходимо выполнить следующие шаги:

1. При активном проекте теплогидравлики вызвать редактор базы данных сигналов через главное меню программы. Пункт меню «Инструменты **→** База данных...» (Рисунок 20).

Рисунок 20. Вызов редактора базы данных для теплогидравлики аналогичен вызову из схемы автоматики

Поскольку данная схема будет присоединена к уже существующей базе данных, необходимо запретить автоматическое сохранение базы данных при сохранении схемы теплогидравлики. Поэтому необходимо перейти на вкладку «Настройки» и снять галочки в пунктах «Сохранять базу» и «Резервировать БД» (

1. Рисунок 21), если галочки там стоят.
2. Выключив опции, переключиться на закладку «Редактор» и закрыть «Редактор базы данных», нажав «Ок».
3. Сохранить схему и закрыть проект (закрыть проект необходимо т.к. его нужно заново загрузить с прописанной базой сигналов, чтобы она считалась из файла, и в проекте теплогидравлики стали доступными сигналы, созданные ранее через схему автоматики).

|  |  |
| --- | --- |
| Рисунок 21. Настройки сохранения базы сигналов для теплогидравлической схемы | Данные установки позволяют не сохранять базу данных при сохранении проекта (файла) схемы теплогидравлики. Это позволяет исключить ошибочное изменение базы данных при редактировании схемы.  Если все вышеописанные действия были выполнены правильно, то после повторного открытия теплогидравлической схемы база данных сигналов автоматически загрузится из файла signals.db и будет содержать все сигналы, сформированные при создании схемы автоматики. |

# Создание простой теплогидравлической модели

## Создание расчетной теплогидравлической схемы

Откройте файл с именем Схема теплогидравлики 1.prt. Данный файл был создан при выполнении второго учебного задания и настроен на работу с базой данных, сохраненной в файле signals.db.

Убедитесь, что база данных содержит сигналы, созданные при выполнении первого учебного задания.

Для создания схемы используются блоки, расположенные в закладке «Теплогидравлика» палитры блоков. (Рисунок 22). Здесь собраны расчетные блоки, непосредствено из которых набираются теплогидравлические модели для расчета кодом HS (HydroSolver). Расчётный код HS входит в стандартную поставку ПО SimInTech, является одномерным односкоростным кодом и предназначен для расчета теплогидравлических контуров произвольной топологии в приближении сжимаемой или несжимаемой среды. Среди доступных теплоносителей – вода и водяной пар, воздух, аргон, жидкометаллический свинец, авиационной топливо и ряд других.

Применительно к учебной задаче будет создана простая модель горизонтального трубопровода между двумя точками с постоянным давлением. На трубопроводе будут размещены две задвижки и точка контроля, измеряющая давление в среднем узле. В качестве учебного задания необходимо будет реализовать простую модель двигателей и блоков управления задвижками, а также алгоритм регулятора, поддерживающего давление в среднем узле на заданном уровне путём формирования управляющих воздействий на первую задвижку. Для второй задвижки будет создан пульт «местного» управления для управления задвижкой вручную.

Рисунок 22. Библиотека расчетных блоков теплогидравлики

Для набора схемы необходимо выполнить действия:

1. Поместите на схемное окно следующие расчетные теплогидравлические блоки (Рисунок 23) :

– «**HS - Граничный узел**»;

– «**HS - Канал**»;

– «**HS - Внутренний узел**»;

– «**HS - Канал**»;

– «**HS - Граничный узел**»;

Рисунок 23. Нужные блоки из библиотеки расчетных блоков теплогидравлики

Изображение блоков схемы может состоять из нескольких связанных графических элементов, которые могут быть перемещены относительно друг друга. Пример – Рисунок 24:

Рисунок 24. Тестовая схема теплогидравлической модели – набор несоединенных блоков.

1. Произведите последовательное соединение элементов таким образом, чтобы два элемента «**HS - Канал**» образовали одну гидравлическую линию с внутренним узлом и граничными узлами. Граничные узлы будут определять давление на границах данной гидравлической линии (Рисунок 25).

Рисунок 25. Тестовая схема теплогидравлической модели – соединение каналов.

1. Поместите на первый элемент «**HS - Канал**» элемент «HS - Клапан» из группы блоков «**HS - Арматура**».
2. Поместите на второй элемент «**HS - Канал**» элемент «HS - Клапан» из группы блоков «**HS - Арматура**».
3. В итоге схема теплогидравлической модели должна выглядеть сходно с рисунком (Рисунок 26).

Рисунок 26. Тестовая схема теплогидравлической модели.

Примечание: для целей настоящего упражнения не важно, будет ли арматура называться «Клапан» или «Задвижка» т.к. подробное рассмотрение блоков управления для них не предусмотрено. Мы выбрали клапан т.к. у него в составе библиотечного блока выведено значение текущей позиции на схему, что удобно при отладке.

## Настройка параметров расчетной модели

Для корректного расчета теплогидравлической модели необходимо задать свойства **каждого** блока (элемента) схемы. Для этого необходимо:

1. Выделить нужный элемент;
2. Во всплывающем меню выбрать пункт «Свойства объекта» (Рисунок 27).

Рисунок 27. Контекстное меню элемента схемы

После этого появится диалоговое окно «Свойства», в котором можно задать свойства элемента. Ниже представлено диалоговое окно для блока «**HS -** Граничный узел» (Рисунок 28 для граничного узла, расположенного слева).

|  |  |
| --- | --- |
|  |  |
| Рисунок 28. Диалоговое окно «Свойства» для левого граничного узла | Рисунок 29. Диалоговое окно «Свойства» для правого граничного узла |

Задайте в граничных узлах значения следующих свойств:

* + - 1. Для левого узла (Рисунок 28):
         * Давление – **1.5e5**;
         * Энтальпия и Температура - оставить заданное по умолчанию;
         * Теплоноситель – **Вода**; (или оставить пустым, по умолчанию будет применена Вода)
      2. Для правого узла (Рисунок 29):
         * Давление – **1.0e5**;
         * Энтальпия и Температура - оставить заданное по умолчанию;
         * Теплоноситель – **Вода**; (или оставить пустым, по умолчанию будет применена Вода)

Этими свойствами будет задана постоянная величина перепада давления (половина атмосферы) на границах моделируемого трубопровода, что приведет к наличию расхода в трубопроводе при незакрытых задвижках.

Обратите внимание, что свойство «**Проходное сечение**» для обоих граничных узлов задано по умолчанию интерпретируемым выражением. Разберем значение для **S**, равное **pi\*Self.D^2/4** для понимания:

* Pi – встроенная в интерпретатор SimInTech константа, численно равная π;
* **Self** – указатель блока на себя самого, т.е. указатель того, что будет использовано в выражении свойство или параметр самого блока;
* **Self.Dg** – свойство Dg данного блока (будет подставлено его значение);
* **^2/4** – возведение во вторую степень и деление на 4.

Таким образом, достаточно задать значение для гидравлического диаметра, а свойство для проходного сечения будет вычислено по формуле для круглой трубы (0.25\*π\*d2/4). При необходимости заданные по умолчанию значения надо изменить на требуемые, как мы это сделали для давления в левом узле.

Если мы хотим в формировании каких-либо одних свойств блока использовать какие-либо другие свойства блока, то нам достаточно сформировать на них ссылку по шаблону **Self.<имя свойства>** и добавить какие-либо математические действия. При создании параметризованных субмоделей часто вместо служебного слова **Self** используется **Submodel** для ссылки на свойство субмодели.

Заполните свойства для блоков «**HS - Канал**» в соответствии с рисунком (Рисунок 30), вызвав диалоговое окно «Свойства».

|  |  |
| --- | --- |
| Рисунок 30. Диалоговое окно «Свойства» для канала. | Для корректной работы задвижек в схеме необходимо, чтобы гидравлическое сопротивление участка трубопровода, в котором они установлены, не равнялось нулю, поэтому нужно задать общее сопротивление канала, равное хотя бы 1 (как для левого трубопровода, так и для правого). Для этого в строках «Прямое местное сопротивление» и «Обратное местное сопротивление» введите **1.** Если количество расчетных элементов (N) больше одного (т.е. канал разбит на ячейки), то сопротивления можно задать строкой Self.N#(1/Self.N), то есть формируется массив из значений сопротивления для каждой гидравлической связи внутри канала так, чтобы суммарное сопротивление канала было равно 1.  По умолчанию задано верное значение. Численное значение сопротивления в данном упражнении не важно, главное чтобы оно было ненулевое. |

Теперь перейдём к задвижкам. Напомним, что при создании базы данных мы использовали название для задвижек Z1 и Z2.

## Изменение названий клапанов на схеме

Помещенные на схему клапаны по умолчанию имеют названия, соответствующие некоторой кодировке оборудования. Средства программного комплекса позволяют задавать шаблон названий по умолчанию для любых элементов. В данном учебном задании мы используем названия «Z1» и «Z2» и не делаем различия между клапанами и задвижками. Прежде чем перейти к выполнению следующего учебного задания, измените названия задвижек.

Для изменения названия клапанов следует выполнить следующие действия:

|  |  |
| --- | --- |
| Рисунок 31. Изменение наименования клапана | 1. Выделите клапан на схеме; 2. Правой кнопкой мыши вызовите меню, в котором выберите пункт «Свойства объекта»; 3. В появившемся окне «Свойства» перейдите на закладку «Общие»; 4. В строке «**Имя объекта**» введите Z1 (Рисунок 31); 5. Нажмите кнопку «**Ок**» для сохранения изменений.   Аналогичным образом измените имя блока (объекта) для второго клапана, на Z2. |

## Связь свойств расчетных элементов с сигналами из базы данных

После создания простейшей расчетной схемы необходимо связать свойства расчетных элементов с сигналами из базы данных. В настоящем упражнении мы свяжем сигналы положения задвижек из базы данных и соответствующие свойства двух блоков-задвижек расчетной схемы.

Выделите на схеме блок «**HS - Задвижка**» и вызовите для него диалоговое окно «**Свойства**» (Рисунок 32):

Рисунок 32. Диалоговое окно редактирования свойств элемента «Регулирующий клапан»

Параметры, представленные в окне редактирования на закладке «Свойства» (Рисунок 32), могут быть заданы несколькими способами:

* с помощью числовых значений непосредственно в редакторе свойств;
* с помощью интерпретируемых математических выражений непосредственно в редакторе свойств;
* посредством импорта из базы данных сигналов – в этом случае в качестве интерпретируемого значения подставится имя сигнала из базы данных (его можно вписать и вручную!) и при его интерпретации (на каждом шаге расчета если сигнал имеет способ расчета «переменная») свойство будет принимать текущее значение сигнала. Таким образом в теплогидравлический код будет передаваться обновляемое значение сигнала из базы данных на каждом шаге синхронизации.

Чтобы связать свойство объекта с сигналом из базы данных, необходимо:

1. В окне редактора свойств выделить значение свойства (в данном примере – число 100, заданное по умолчанию, у свойства «Степень открытия, %»);
2. Нажать кнопку «Найти значение в базе» (Рисунок 32). При нажатии этой кнопки происходит вызов диалогового окна «Редактор базы данных» (Рисунок 33). В диалоговом окне нужно выбрать последовательно:

* Категорию **«Задвижки»**;
* Группу сигналов **«Z1»** для левой задвижки;
* Имя сигнала **«Положение»**.

1. Выберите данный сигнал и нажмите клавишу «Добавить» в панели «Выбранные данные» (Рисунок 33). Для выбранного сигнала формируется уникальное имя, состоящее из имени группы сигналов и имени сигнала, разделенных знаком подчеркивания (для данной задвижки имя сигнала будет Z1\_xq1);
2. Введите значение 50 для сигнала Положение (Рисунок 33). Задание сигнала в базе данных устанавливает начальное значение положения задвижки. В данном случае это значит, что задвижка открыта на 50%.
3. Закройте окно редактора базы данных нажатием кнопки «Ок».

Рисунок 33. Выбор сигнала в базе данных для свойств объекта

Повторите действия для второго клапана, привязав его состояние к аналогичному сигналу из базы данных проекта, но относящемуся к группе сигналов с именем «Z2». Диалоговое окно свойств этой задвижки должно выглядеть так, как показано на рисунке (Рисунок 34). Имя сигнала **Z2\_xq01** можно вписать и вручную, так бывает удобно поступать, когда разработчик модели хорошо знает базу сигналов и именования сигналов не длинные. Для надёжности лучше вставлять имя сигнала из базы.

Рисунок 34. Свойства клапана после связывания сигнала Z2\_xq01 из базы данных и «Положение, %»

**Подчеркнём еще раз**: вышеописанный способ не является единственным способом связи какого-либо сигнала из базы данных и свойства блока. Кроме того что Пользователь может вводить имя используемого сигнала вручную (в нашем случае в свойстве задвижек «Степень открытия, %» можно было ввести **Z1\_xq01** и **Z2\_xq01** соответственно, вообще не вызывая редактор базы данных), можно также присваивать значения свойств через Скрипт проекта или субмодели. В некоторых ситуациях бывает нужно переопределять само интерпретируемое значение в процессе расчета, однако эти более сложные ситуации не входят в рамки настоящих учебных заданий.

## Просмотр расчетных параметров теплогидравлической схемы

Каждый блок расчетной схемы содержит набор параметров, которые рассчитываются кодом и отражают текущее состояние расчетных характеристик процесса, происходящего в схеме и блоке. Контролируя параметры блока как в числовом, так и в графическом виде, пользователь может анализировать ход переходного процесса. Используя параметры, пользователь может создать точки контроля для блока, которые в свою очередь могут использоваться в схемах автоматики для формирования различных алгоритмов управления (например, параметр «давление» - алгоритм управления «поддержание давления на заданном уровне»). Эти параметры, как в виде табличных значений, так и виде графиков, можно просматривать непосредственно во время расчета теплогидравлической схемы. Для того, чтобы просмотреть список параметров, доступных для каждого элемента схемы, необходимо:

1. Выделить элемент теплогидравлической схемы;
2. Нажать правую кнопку мыши.
3. Во всплывающем меню выбрать пункт «Параметры объекта» (Рисунок 35).

Рисунок 35. Пункт «Параметры объекта» в контекстном меню

После этого появляется окно со списком параметров, которые можно получить из расчетного кода для данного элемента схемы. Это же окно может быть вызвано двойным щелчком по блоку в процессе расчета, если схема находится в режиме «Редактирование» (НЕ в режиме «Индикация»).

Приведем данный алгоритм для блока «**HS – Внутренний узел**». Выделите нажатием левой кнопки мыши на теплогидравлической схеме данный блок и вызовите окно параметров.

Появится диалоговое окно «**Параметры**» для выбранного блока (в данном случае для блока «**HS – Внутренний узел**», Рисунок 36). Окно отображает список параметров блока, выбранного в данный момент на схеме. Не закрывая это окно, можно выделить другой элемент схемы и посмотреть список его параметров.

|  |  |
| --- | --- |
| Рисунок 36. Окно «Параметры» для внутреннего узла схемы | Рисунок 37. Окно временного графика |

Выберите в списке параметр «Давление» и нажмите здесь же, слева вверху, кнопку «Создать график» (Рисунок 36). Появится новое окно «Временной график», в котором будет отображаться изменение выбранного параметра по времени (Рисунок 37).

Имя переменной на графике формируется из имени элемента (данное имя можно увидеть, вызвав окно «**Свойства**» и открыв вкладку «**Общие**») и имени параметра, которые разделяются точкой. Например, для элемента «Внутренний узел», с именем «Node\_2.», и параметра давление с наименованием «\_p», имя переменной на графике – «Node\_2.\_p» (Рисунок 37).

Не закрывая диалогового окна «Параметры», выберите на теплогидравлической схеме первый (левый) элемент «HS - Канал». Диалоговое окно параметры при этом будет отображать список параметров, соответствующих уже вновь выбранному элементу «HS - Канал» (Рисунок 38).

|  |  |
| --- | --- |
| Рисунок 38. Окно «Параметры» для канала | Выделите параметр «\_g (Массовый расход, кг/с)» и нажмите кнопку «Создать график».  Запустите задачу на расчет (пункт «Расчёт → Пуск» Главного меню). Запустить задачу на расчет также можно нажав клавишу «F9» на клавиатуре либо используя кнопки запуска расчета из схемного или главного окна.  Если предыдущие действия выполнены правильно, после короткого переходного процесса в созданной схеме установятся следующие значения для выбранных параметров:  Давление во внутреннем узле = 125000 (Рисунок 39), массовый расход по трубопроводу примерно 25,1 кг/с (Рисунок 40).  При автономном моделировании теплогидравлической модели (без автоматического управления) пользователь имеет возможность просматривать любые параметры любого расчетного блока на схеме и выводить параметры на график. |

Мы сделали график для давления во внутреннем узле, расход по каналу предлагается вывести на график самостоятельно. Также есть возможность оказывать влияние на некоторые другие величины – граничные условия, положение задвижек можно менять либо вручную (если положение задано числом), либо через базу сигналов, если положение задано сигналом.

|  |  |
| --- | --- |
| Рисунок 39. График давления во внутреннем узле схемы | Рисунок 40. График массового расхода в трубе |

Для примера, вы можете изменить давление в левом граничном узле с 1.5e5 на какое-либо другое (1.4e5, 1.1e5 или 1.7e5) и посмотреть как это отразится на давлении в среднем узле и на массовом расходе. Следует учесть что подобные изменения происходят (для расчетного кода) мгновенно и важно не слишком сильно «дёргать» расчетную схему, чтобы она оставалась устойчивой для численного интегрирования. Другой пример – можно изменить гидравлический диаметр одного из каналов и это скажется также как на итоговом расходе, так и на давлении в среднем узле. Если, например, задать давление слева таким же как и справа то течение теплоносителя через трубопровод прекратится…

## Добавление датчиков (точек контроля) к теплогидравлическим элементам

На теплогидравлической схеме можно разместить дополнительные блоки – датчики, которые позволяют считывать параметры, рассчитываемые различными блоками, и записывать их с каким-то технологическим именем в базу данных. Каждый элемент расчетной схемы содержит набор параметров, которые можно передавать в базу данных сигналов с помощью датчиков.

Передача параметров через базу данных позволяет создавать математическую модель системы управления, получающую сигналы от датчиков из теплогидравлической модели, в полном соответствии с реальным объектом.

Разместим на схеме датчик давления во внутреннем узле схемы. Для этого необходимо выполнить следующие действия:

1. Вставьте блок «HS – Датчик P в узле» в элемент «Внутренний узел».
2. В свойствах блока в поле «**Имя категории для точек контроля»** задайте строку «**Датчики**»
3. Задайте в поле «Имя точки контроля» строку «D1».
4. Задайте в поле «Значение (Источник)» строку «parent.\_P».
5. Задайте в поле «**Сигнал (Приемник)**» строку «**iq01**».
6. Задайте в поле «Название сигнала» строку «Расчетное давление в узле, МПа» (Рисунок 41).

Таким образом, мы добавили на схему датчик, который «измеряет» давление во внутреннем узле. По сути, этот блок на каждом шаге расчёта будет считывать значение параметра **\_p** у внутреннего узла и записывать значение в сигнал **D1\_iq01** базы данных.

**Важное примечание**: у точек контроля есть общее свойство с названием «Имя категории для точек контроля». Его необходимо изменить на «**Датчики**» (Рисунок 42) т.к. в нашей базе сигналов именно в этой категории хранятся сигналы для датчиков.

|  |  |
| --- | --- |
| Рисунок 41. «Точка контроля» давления во внутреннем узле  Запустите задачу на расчет нажатием на пункт Главного меню «Расчёт → Пуск». Если добавление датчика на схему было выполнено правильно, то в базе данных в категории «Датчики» в группе сигналов «D1» будет видно изменение текущего значения давления в среднем узле. | Рисунок 42. Общие свойства «Точки контроля» |

Вызовите редактор базы данных (пункт «Инструменты → База данных» Главного меню). Установите в «Редакторе базы данных» галочки в полях «Режим просмотра сигналов» и «Обновлять с интервалом» (Рисунок 43). В этом режиме «Редактор базы данных» отображает текущие значения сигналов, рассчитываемых схемой (а не начальные, задаваемые пользователем при редактировании базы данных). Убедитесь, что значение в таблице для сигнала **iq01** группы сигналов D1 соответствуют давлению в узле схемы.

Рисунок 43. Редактор базы данных в режиме просмотра значений сигналов для точки контроля.

Отметим, что созданные ранее сигналы для датчиков D2 и D3 нами не использованы, но могут быть задействованы при дальнейшей разработке схемы. Также, при коллективной работе, сначала как правило создается база сигналов, а уже потом к каждому объекту подключаются «реальные» данные. Это позволяет создавать алгоритмы управления без модели, используя уже «как бы» готовые датчики, а разработчик модели позже «привязывает» к датчикам те или иные расчетные параметры из модели. Либо наоборот, если модель создается ранее, показания точек контроля «висят» в базе данных, но еще никем не используются.

# Создание простого алгоритма управления

## Создание простейшего алгоритма управления

Откройте снова файл с именем Схема автоматики 1.prt. Данный файл был создан при выполнении первого учебного задания и настроен на работу с базой данных, содержащейся в файле signals.db. Убедитесь, что база данных содержит сигналы, созданные при выполнении первого учебного задания, а также группы сигналов **D1**, **D2** и **D3**, созданные и сохраненные при создании категории **Датчики**.

Прейдите на закладку «Субструктуры» палитры блоков в главном окне программы и выберите блок «Субмодель» (Рисунок 44).

Рисунок 44. Выбор блока «Субмодель» в палитре блоков

Поместите выбранный блок на схемное окно (Рисунок 45). Произведите двойной клик на блоке «Субмодель». Это действие приводит к раскрытию внутренней структуры блока (или переходу на следующий уровень вложенности). Поскольку новый блок пока не содержит никаких элементов, открывается пустое схемное окно (Рисунок 46). Заголовок окна после перехода в субмодель содержит в себе имя файла проекта и имя субмодели, в которой в данный момент находится пользователь. В данном примере это Схема автоматики 1.prt/ Macro8 (Рисунок 46).

|  |  |
| --- | --- |
| Рисунок 45. Схема автоматики с вставленным блоком «Субмодель» | Рисунок 46. Схемное окно внутренней структуры блока «Субмодель» |

Внутри субмодели мы создадим простой алгоритм автоматического управления задвижкой, поддерживающий давление в среднем узле, включая и модель двигателя в виде интегратора. Для этого на м требуется считать из базы сигналов показания точки контроля (давление в среднем узле), обработать его, сравнив с заданным давлением, сформировать сигнал на двигатель, и вычисленное по модели двигателя положение задвижки передать обратно в базу сигналов (чтобы оно далее было использовано в схеме теплогидравлики).

|  |  |
| --- | --- |
| Рисунок 47. Блоки «Чтение сигналов из списка» и «Запись сигналов в список» | Поместите на схемное окно (внутри субмодели) два блока «Запись сигналов в список» и один блок «Чтение сигналов из списка» из линейки блоков «Данные» палитры блоков главного окна программы, разместив их таким образом, как на рисунке (Рисунок 47).  Обратите внимание на букву М – блокам пока не присвоены имена сигналов которые они считывали и записывали бы в базу сигналов. Это предстоит сделать.  Блоки настроены таким образом что они отображают имя того сигнала который читают/записывают, если имя сигнала не очень длинное (от 1 до 20 символов). Иначе блок будет снова отображать букву М. |

## Соединение блоков чтения и записи с сигналами из базы данных

Данные блоки будут осуществлять передачу данных из алгоритма управления в базу данных сигналов (блоки записи), и обратно (блок чтения. Для соединения блока с конкретным сигналом из базы данных необходимо вызвать диалоговое окно «**Свойства**». В появившемся окне необходимо выделить строку «Имена сигналов» и нажать кнопку «Заполнить из базы данных», которая расположена внизу диалогового окна (Рисунок 48).

Рисунок 48. Окно «Свойства объекта» для блока «Запись сигнала в список»

Нажатие кнопки приводит к вызову диалогового окна «Редактор базы данных» (Рисунок 49) в режиме экспорта сигналов. В данном диалоговом окне происходит выбор сигнала (или сигналов), находящегося в базе данных, и подстановка его названия в интерпретируемое значение свойства блока, находящегося на расчетной схеме модели системы управления. Данная привязка аналогична привязке, выполненной у блоков «**HS – Задвижка**» в теплогидравлической модели в предыдущем учебном задании.

В данном учебном задании необходимо связать свойства «Имена сигналов» (у блоков чтения и записи сигналов), сигнал «**Значение из модели**» для датчика **D1**, и сигналы «Положение» в базе данных для задвижек Z1 и **Z2** (Рисунок 49, пример для задвижки Z1). В окне редактора базы данных надо выбрать нужный сигнал, предварительно выбрав категорию и группу сигналов, нажать кнопку «Добавить» в панели «Выбранные данные» (Рисунок 49).

Рисунок 49. Выбор сигнала в базе данных для свойств объекта

Имя сигнала, выбранное в диалоговом окне, отражается на схемном окне в рамках соответствующего блока. Настройте аналогично второй блок «Запись сигналов в список» используя сигнал «**Положение**» для задвижки с именем группы сигналов «Z2».

Блок «Чтение сигналов из списка» свяжите сигналом «Давление в узле» из базы данных, относящимся к категории «Датчики» и группе сигналов D1.

Если все действия выполнены правильно, то надписи на блоках должны выглядеть примерно так, как показано на следующем рисунке (Рисунок 50):

Рисунок 50. Субмодель после выполнения связи блоков с базой сигналов

## Создание простейшей схемы управления

Поместите на схему дополнительно блок «Константа» из закладки «Источники». Установите значение свойства блока равным «117000» (этот блок будет «задатчиком» давления в узле, которое должен «держать» регулятор). Свойство «**Название**» - информационное, служит для оформления блока-константы на схеме.

|  |  |
| --- | --- |
| Поместите на схему блок «Ступенька» из закладки «Источники». Установите следующие значения для свойств блока (Рисунок 51):   * Время срабатывания **100** (с). * Начальное состояние **50** (%). * Конечное состояние **10** (%). | Рисунок 51. Свойства блока типа «Ступенька» |

Поместите на схему блок «Сравнивающие устройство» из закладки «Операторы». У этого блока по умолчанию заданы весовые коэффициенты по входам равные +1 и -1, т.е. блок работает в режиме простого вычитания сигналов и на выходе формирует разность между первым и вторым входным сигналом (рассогласование между измеренным давлением и заданным в нашем случае).

|  |  |
| --- | --- |
| Поместите на схему блок «Интегратор с ограничением» из закладки «Динамические». Установите следующие значения для свойств блока (Рисунок 52):   * Коэффициент усиления: **-0.001;** * Максимальное значение: **100;** * Минимальное значение: **0;** * Начальные условия: **50.** | Рисунок 52. Свойства блока типа «Интегратор с ограничением» |

Свяжите блоки линиями связи в структурную схему, как показано следующем рисунке (Рисунок 53):

Рисунок 53. Схема алгоритма регулирования, модели двигателя задвижки Z1 и программного управления задвижкой Z2

Сохраните созданный проект.

Созданная схема содержит простой алгоритм регулирования, который обеспечивает поддержание давления на уровне 117 кПа во внутреннем узле за счет открытия/закрытия задвижки на первом участке трубопровода.

Сигнал «Давление в узле» будет передаваться из теплогидравлической модели в базу данных, а затем считываться из базы данных в схему алгоритма управления, в которой он сравнивается с константой 117000 (заданное давление в Па). Результат данного сравнения подается на интегратор с ограничением. Если давление больше заданного, то на выходе из сравнивающего устройства формируется положительное рассогласование, а за счет отрицательного коэффициента усиления интегратора (коэффициент усиления -0.001) значение сигнала на выходе интегратора начинает уменьшаться (т.е. происходит эмуляция закрытия «задвижки»), если давление меньше заданного, то формируется отрицательное рассогласование и значение сигнала на выходе интегратора начинает увеличиваться (т.е. происходит эмуляция открытия «задвижки»).

Положение второй задвижки формируется программным образом при помощи блока «**Ступенька**»: в начальный момент времени положение задвижки имеет значение 50 %, а затем меняется скачком до 10 (%) через 100 секунд после начала расчета.

## Проверка обмена с базой данных сигналов

Созданная схема алгоритма управления в режиме расчета обменивается сигналами с базой данных. Так как на данном этапе к базе не подключена теплогидравлическая модель, то можно проверить только процедуры записи и чтения сигналов. Это так называемый режим «автономной отладки» алгоритма.

Произведите инициализацию схемы: пункт меню «Расчет» главного меню программы, подпункт «Инициализация» (Рисунок 54) либо используя кнопку «**Инициализация**», расположенную в панели кнопок управления расчетом в схемном или главном окне.

Рисунок 54. Инициализация расчета

Войдите в редактор базы данных и убедитесь, что положение задвижек в начальный момент времени равно 50. Запустите схему на расчет.

Во время расчета сигнал давления, содержащийся в базе данных и используемый в алгоритме, равен нулю. Соответственно, первая задвижка открывается практически сразу, и ее положение увеличивается до 100%. Вторая задвижка приходит в положение 50. Через 100 секунд расчета «срабатывает» блок «Ступенька», и вторая задвижка переходит в положение 10.

Для отслеживания изменений текущих значений сигналов в базе данных необходимо установить галочки в пунктах «Режим просмотра текущих значений» и «Обновлять с интервалом» (Рисунок 55). Со снытой галочкой «Режим просмотра текущих значений» интерфейс базы отображает начальные значения сигналов.

Рисунок 55. Редактор базы данных в режиме просмотра значений сигналов

# Создание комплексной модели

Комплексная модель – это, как правило, сложная модель объекта, содержащая в своем составе несколько простых моделей. Комплексная модель может быть сформирована из математических моделей, рассчитываемых как одинаковыми, так и разными решателями (встроенными в SimInTech, или сторонними расчетными кодами, или другими средствами). В комплексной модели обеспечивается совместный синхронный расчет составляющих ее моделей. Для обмена данными между моделями используется единая база данных сигналов.

В терминах файлов SimInTech каждая модель является файлом проекта (проектом), и файл имеет расширение **prt**. Комплексная модель – это пакет проектов. Файл пакета имеет расширение **pak**. Пакет проектов может быть запущен на расчет как на локальном компьютере, так и распределён по вычислительной сети. В этом случае говорят о «распределённом пакете». В настоящем руководстве описан процесс создания пакета проектов и его запуск на расчет локально, на одном рабочем месте.

## Создание нового пакета для комплексной модели

Выполняя предыдущие учебные задания, мы создали две простейшие математические модели: модель теплогидравлики (файл Схема теплогидравлики 1.prt) и модель системы управления (файл Схема автоматики 1.prt), которые используют одну и ту же базу данных сигналов (файл signals.db).

Для создания комплексной модели будет использован новый тип файла среды SimInTech, который называется «**Пакет**».

Выполните следующие действия:

1. Закройте все открытые в SimInTech проекты (при добавлении проектов в пакет они будут автоматически открыты, но уже в составе пакета);
2. Выберите в Главном меню пункт **«Файл → Новый проект → Пакет»** (Рисунок 56).

|  |  |
| --- | --- |
| Рисунок 56. Меню создания комплексной модели | Рисунок 57. Окно управления пакетом проектов |

1. Появится окно **«Пакет проектов»**, которое служит для создания и управления комплексной математической моделью (Рисунок 57).

Данное окно разделено на три области:

* 1 область – панель инструментов (кнопок). В данной области содержатся кнопки для сохранения файла пакета, добавления и удаления проектов, управления расчетом и настройки пакета;
* 2 область – окно проектов. В данном поле будет представлен список проектов, которые включены в пакет. В новом пакете область пуста.
* 3 область – окно сообщений. В данной окне появляются различные сообщения, сообщающие о ключевых событиях, связанных с инициализацией, расчетом, загрузкой/сохранением рестартов и остановкой пакета проектов.

1. Нажав на кнопку «Сохранить пакет», сохраните пакет под именем, например, **Pack1.pak** в том же каталоге, где расположены файлы **Схема теплогидравлики 1.prt** и **Схема автоматики 1.prt** (обратите внимание что заголовок окна пакета изменится).
2. В окне «Пакет проектов» нажмите кнопку **«Добавить проект»**. В появившемся окне выберите файл **Схема теплогидравлики 1.prt** с гидравлической моделью, созданной на предыдущих занятиях;
3. Добавьте в пакет второй проект – файл **Схема автоматики 1.prt**.

Каждый добавляемый проект автоматически открывается в SimInTech. Работа (с точки зрения пользователя) с проектами, открытыми таким образом, ничем не отличается от работы с проектами, открытыми по отдельности. Пользователь может переключаться между окнами, а также осуществлять редактирование и запуск каждого проекта в отдельности. Закрыть проект, входящий в пакет, отдельно от пакета нельзя. Проекты закрываются вместе с закрытием пакета. Это связано как с удобством работы, так и с особой инициализацией базы сигналов и последовательной инициализацией проектов при запуске на расчет (проекты, входящие в пакет, используют общую базу сигналов, т.е. общую область памяти, и должны при запуске пакета на расчет инициализироваться последовательно, и далее вести расчет синхронно, под управлением общего синхронизатора).

## Расчет комплексной модели

После добавления проектов окно «**Пакет проектов**» должно принять следующий вид (Рисунок 58).

Запустите пакет проектов, используя кнопку «Запустить все» в окне управления пакетом (изображение аналогично кнопке Пуск Главного Окна, Рисунок 58). При нажатии на данную кнопку происходит запуск на расчет математических моделей автоматики и теплогидравлики, входящих в пакет.

|  |  |
| --- | --- |
| Подождите пока модельное время достигнет 30-50 секунд и нажмите на кнопку «Пауза» в окне «Пакет». За модельным временем можно наблюдать в любом из схемных окон. При этом происходит приостановка расчета во всех математических моделях, входящих в файл пакета.  Поскольку открытые проекты использую общую базу данных сигналов, то происходит сквозной обмен данными между моделями: каждый из проектов на каждом расчетном шаге обращается к базе данных для чтения и/или записи сигналов. | Рисунок 58. Окно управления пакетом проектов |

В созданном примере модель автоматики получает из базы данных значение давления в промежуточном узле из теплогидравлической модели и воздействует на первую задвижку таким образом, чтобы свести к минимуму рассогласование текущего и заданного давлений, т.е. поддерживать давление в узле на уровне 117кПа. В промежуток времени между 30 и 50 секундой расчета переходные процессы в модели завершаются, и устанавливается состояние, близкое к стационарному.

Нажмите кнопку «Запустить все», продолжив тем самым расчет пакета. Наблюдайте за ходом расчета.

В нашей модели на 100-й секунде расчета происходит мгновенное изменение положения второй задвижки с 50 до 10 за счет срабатывания блока «Ступенька» в модели автоматики, что приводит к скачкообразному повышению давления в промежуточном узле.

Простейший регулятор, созданный в модели автоматики, прикрывает первую задвижку с тем, чтобы обеспечить заданное давление в узле. В итоге положение первого клапана устанавливается на уровне 7,2% (Рисунок 59).

Рисунок 59. Схемное окно теплогидравлической модели, время 185 сек

|  |  |
| --- | --- |
| Скачок давления на сотой секунде расчета виден на графике давления, который был создан при работе с теплогидравлической моделью (Рисунок 60).  Положение задвижек приведено на рисунках ниже (Рисунок 61 и Рисунок 62Рисунок 62). Остановите расчет по достижении модельного времени 150-200 секунд. | Рисунок 60. График давления во внутреннем узле |

Рисунок 61. График положения задвижки Z1

Рисунок 62. График положения задвижки Z2

# Создание блока управления оборудованием

## Создание следующего проекта автоматики

В предыдущих учебных заданиях была создана комплексная модель, состоящая из модели теплогидравлики и модели автоматики, обменивающихся между собой данными при помощи общей базы данных сигналов.

В данном задании будет продемонстрирована возможность создания типовых блоков управления средствами SimInTech. При моделировании сложных систем часто возникает ситуация, когда необходимо использовать одну и ту же типовую математическую модель многократно. В SimInTech можно использовать одну математическую модель для нескольких однотипных объектов с помощью механизма векторной обработки сигналов.

Для демонстрации возможностей SimInTech мы используем уже существующую базу данных, созданную при выполнении предыдущих заданий. В качестве простейшего примера мы создадим типовой алгоритм управления задвижкой (включая и модель двигателя).

|  |  |
| --- | --- |
| 1. Откройте файл **«Схема автоматики 1.prt»**, созданный при выполнении предыдущих заданий; 2. Удалите существующую **«Субмодель»**; 3. Сохраните файл под новым именем **«Схема автоматики 2.prt»,** используя команду «**Сохранить проект как…**», которая находится в меню «**Файл**» (Рисунок 63). | Рисунок 63. Меню сохранения файла под новым именем |

Поскольку ранее созданный файл был связан с базой данных, файл с новым именем также будет связан с этой базой данных.

В отличие от предыдущего примера, когда в системе автоматики рассчитывалось положение задвижки, и оно непосредственно передавалось в модель теплогидравлики, мы попытаемся создать более приближенную к реальности математическую модель задвижки. Новая модель задвижки будет получать команды на открытие и закрытие и, согласно получаемым сигналам, формировать команду на двигатель и менять положение. Также «блок управления» будет формировать признаки состояния задвижки – открыта она или закрыта.

## Добавление новых сигналов в базу данных

Команды на открытие и закрытие задвижек будут предаваться через базу данных сигналов, поэтому необходимо внести изменения в существующие в базе данных категории. В учебном примере мы используем категорию «Задвижки» для обмена данными с клапанами.

Приступим к редактированию:

1. Войдите в **«Редактор базы данных»**. Пункт меню **«Инструменты → База данных**».
2. В редакторе выберите категорию **«Задвижки»** и нажмите кнопку **«Настроить категорию»** (Рисунок 64), либо выполните двойной щелчок мыши по категории **«Задвижки»**.

Рисунок 64. Редактор базы данных

1. В окне **«Свойства категории»** добавьте два новых сигнала (кнопкой **«Добавить сигнал»** в нижней части окна): **«Команда Открыть»** и **«Команда Закрыть»,** как показано на рисунке ниже(Рисунок 65).

Рисунок 65. Настройка категории

1. Закройте окно **«Свойства категории»** нажатием кнопки **«Ок»**.
2. Если теперь выбрать любую из присутствующих в категории «Задвижки» группу сигналов, то в сводной таблице сигналов мы увидим два новых сигнала, которые мы только что добавили; то есть по модифицированному шаблону добавились сигналы и в группы сигналов.
3. Закройте «**Редактор базы данных**» нажатием кнопки «**Ок**».

## Создание блока управления оборудованием

Под блоками управления оборудованием обычно рассматриваются типовые алгоритмы управления, отвечающие за управление типовым оборудованием, установленным на технологическом объекте. К такому оборудованию как правило относятся задвижки различных типов, запорные и регулирующие клапана, двигатели различного назначения и пр. Данное оборудование, обладая различными физическими характеристиками, обычно имеет один и тот же алгоритм управления, соответствующий ее типу: например, большинство задвижек на таком объекте, как АЭС, будет управляться по алгоритму управления задвижками.

При моделировании зачастую совместно с блоком управления моделируется и сам орган управления (исполнительный механизм), если отсутствует отдельная модель электроэнергетической системы, т.к. требуется предусмотреть типовые отказы на оборудование и на блок управления.

В данном блоке мы создадим простейшую модель управления задвижкой с возможностью расчета ее положения: на вход в модель будут подаваться две управляющих команды «Открыть» и «Закрыть», а на выходе мы будем получать положение задвижки и ее состояние: **«Открыта»** или **«Закрыта»**.

Рассмотрим процедуру создания блока управления оборудованием:

1. Поместите на схему новый блок **«Субмодель»** из закладки **«Субструкутры»**.
2. Создадим подпись для данного блока. Это необходимо для того, чтобы в дальнейшем при работе со схемой ясно понимать в какой субмодели находится та или иная часть алгоритма/модели. Для этого вызовите окно **«Свойства»**, и на вкладке **«Общие»** найдите свойство **«Подпись»**. Введите в поле «**Значение**» текст **«Управление оборудованием»**. Сохраните внесенные изменения нажатием на кнопку **«Ок»**. После этого отображение блока «Субмодель» примет следующий вид (Рисунок 66);
3. Подпись можно ввести и льтернативным способом – двойным нажатием под блоком. Там следует ввести (в появившемся поле редактора) название субмодели **«Управление оборудованием»** (Рисунок 66). **Примечание**: субмодель слева либо уже удалена, либо её можно пока оставить и удалить позже. Также можно воспользоваться режимом исключения из расчета блока субмодели, без удаления из схемы.

Рисунок 66. Создание подписи блока

1. Осуществите двойной клик на блоке **«Управление оборудованием»**.
2. Поместите на схему новый блок **«Субмодель»** из закладки **«Субструкутры»** и подпишите ее как **«БУЗ»**.

В данном блоке мы создадим простейшую модель двигателя и блока управления клапана (задвижки). На вход в модель будут две команды «Открыть» и «Закрыть». На выходе мы будем получать положение задкижки, и сигналы её состояния – открыт он или закрыт.

1. Осуществите двойной клик на блоке **«БУЗ»**.
2. Поместите на схему два блока «**Порт входа**» и три блока «**Порт выхода**» из закладки «**Субструкутры**» (Рисунок 67). Данные блоки работают не с базой даных, а организуют входные и выходные порты для субмодели.
3. Осуществите двойной клик на блоке **«Порт входа»**.

Рисунок 67. Блоки «Порт входа» и «Порт выхода» в палитре блоков

Рисунок 68. Редактирование имени порта субмодели

1. В окне **«Порт субмодели»** введите строку **«Команда Открыть»** для первого порта входа и **«Команда Закрыть»** для второго (Рисунок 68).
2. Измените имена блоков портов субмодели так, чтобы схема приняла следующий вид (Рисунок 69).

Рисунок 69. Схема субмодели БУЗ с переименованными портами

Для осуществления перехода на один уровень вложенности выше выполните двойной клик на пустом месте схемы, либо нажмите на зелёную стрелку вверх в панели инструментов схемного окна. Обратите внимание, что после добавления портов в схему субмодели, её изображение на схеме также дополнилось портами, к которым можно подводить линии связи. При наведении курсора на любой порт всплывает подсказка с именем порта, присвоенным внутри модели (Рисунок 70).

Рисунок 70. Блок субмодели БУЗ после добавления портов

## Векторная обработка сигналов

Создаваемый нами блок управления задвижкой будет являться единым для категории «**Задвижки**». Поэтому сигналы, которые должны быть обработаны данным блоком, должны быть векторными, а каждый порт будет обрабатывать столько сигналов, сколько групп сигналов представлено в категории, с которой работает данный блок.

Продолжим наполнение блока **«Управление оборудованием»** и создадим алгоритм, «умеющий» обрабатывать векторный сигнал:

1. Поместите на схему рядом с блоком **«БУЗ»** два блока **«Чтение из списка сигналов»** и три блока **«Запись в список сигналов»**.
2. Для удобства дальнейшей работы поместите на схему также блоки **«Заметка»**, позволяющие ввести подписи на схему (Рисунок 71).

Рисунок 71. Блок «Заметка» в палитре компонентов

1. Осуществите двойной клик на надписи «Текст» блока «Заметка» и введите в окно редактора текст пояснения для каждого блока. Схема должна принять вид представленный на рисунке (Рисунок 72).

Рисунок 72. Схема блока управления оборудованием

Блоки «Чтение сигнала из списка» и «Запись сигнала в список» в предыдущих учебных заданиях использовались для чтения и записи единичного сигнала в базе данных. В данном учебном задании мы будем получать не один сигнал, а массив сигналов по всем задвижкам в базе данных. Для этого необходимо выполнить следующие действия:

1. Осуществите двойной клик на блоке **«Чтение сигнала из списка»**.
2. В появившемся диалоговом окне **«Свойства»** прейдите в строку **«Имена сигналов»** и нажмите кнопку вызова текстового редактора, появляющуюся при редактировании строки и расположенную справа от текста (Рисунок 73).

Рисунок 73. Окно редактирования свойств блока «Чтение сигналов из списка»



Рисунок 74. Текстовый редактор запроса к базе данных

1. В текстовом окне редактора необходимо сформировать запрос к базе данных. Запрос представляет собой текст в виде одной или нескольких строк в фигурных скобках:

{query:

category = "Задвижки";

group = "\*";

name = "yb01"}

, где query: – ключевое слово.

Данный запрос состоит из трех полей:

category = "Задвижки" – название категории («Задвижки»), из которой мы хотим получить сигналы.

group = "\*" – название группы сигналов которую необходимо включить в запрос. Знак \* означает, что в запрос необходимо включить все группы сигналов данной категории. Обратите внимание, что база данных позволяет осуществлять выборку групп сигналов по фильтру.

Например: если в проекте есть задвижки, название которых начинаются на букву D, то для получения сигналов только от таких задвижек достаточно записать group="D\*".

name="yb01" – имя сигнала, который мы хотим получить из базы данных. В данном случае это имя соответствует сигналу «Команда Открыть». Это имя мы задавали, когда формировали свойства категории (Рисунок 65).

1. Закройте текстовый редактор, нажав кнопку **«Применить»** в верхней части окна (Рисунок 74).
2. Закройте окно редактирования свойств блока, нажав кнопку **«Ок»**.
3. Повторите пункты 1–5 для всех блоков чтения и записи сигналов на схеме. При этом следует вводить строки запроса, соответствующие назначению сигналов в базе данных:

«Команда Закрыть»:

{query: category = "Задвижки"; group = "\*"; name = "yb02"}

«Открыта»:

{query: category = "Задвижки"; group = "\*"; name = "xb01"}

«Закрыта»:

{query: category = "Задвижки"; group = "\*"; name = "xb02"}

«Положение»:

{query: category = "Задвижки"; group = "\*"; name = "xq01"}

1. Добавьте на схему блок **«Временной график»** и соедините блоки линиями связи, как показано на рисунке (Рисунок 75):

Рисунок 75. Субмодель управления оборудование после соединения блоков

## Редактирование параметров «нового» блока

Созданный алгоритм позволяет получить массив сигналов (другое название: вектор сигналов) из базы данных по задвижкам, обработать его в блоке **«БУЗ»**, а затем результаты обработки передать обратно в базу данных.

Для корректной работы алгоритма, работающего с вектором сигналов необходимо точно учитывать размерность вектора. Размерность вектора в нашей модели равна количеству групп сигналов, находящихся в категории **«Задвижки»** в базе данных сигналов. В данном учебном задании мы зададим размерность вектора в качестве нового свойства для блока **«БУЗ»**.

SimInTech позволяет создавать новые блоки с произвольным набором свойств и параметров. Созданная субмодель может быть закрыта для редактирования и использована в виде «черного» ящика при разработке более сложных алгоритмов и моделей.

Для добавления нового свойства выполните следующие действия:

1. Выделите **«Субмодель «БУЗ»** на схеме.
2. Выберите в главном окне программы пункт меню «**Правка**» подпункт «**Изменить блок..**» (Рисунок 76). Если данный пункт недоступен, то необходимо включить «**Режим разработчика**», и повторить пункты сначала. В результате появится окно «**Редактор новых блоков**».
3. В окне **«Редактор новых блоков»** нажмите кнопку **«Добавить свойство»** в нижней части окна. Введите следующие значения полей:

|  |  |
| --- | --- |
| Название | Число задвижек |
| Имя | Z\_Count |
| Тип данных | Целое |
| Значение | {query: category = "Задвижки"; group= "\*"; name= "\*xb01"; what=count} |
| Способ расчета | **Переменная** |

(Рисунок 77).

В строке «Значение» формируется запрос к базе данных, который возвращает количество сигналов, в имени которых присутствует текст «xb01» из категории «Задвижки». Поскольку во всех задвижках есть только один сигнал «xb01», то возвращаемое значение соответствует количеству задвижек (клапанов) в базе данных сигналов.

Рисунок 76. Меню изменения блока

1. Закройте диалоговое окно **«Редактор новых блоков»** нажатием кнопки **«Ок»**.

Рисунок 77. Окно редактора новых блоков

## Создание модели управления клапаном

Простейшая модель управления клапаном будет работать по следующему принципу. На вход модели поступают два сигнала логического типа: «Команда Открыть» и «Команда Закрыть». При работе алгоритма возможны несколько вариантов комбинирования данных команд: когда обе команды инициированы, когда обе команды не инициированы, когда инициирована одна из команд. Рассмотрим данные варианты ниже.

В случае, когда обе команды равны «0» (логическая Ложь) или обе команды равны «1» (логическая Истина, но команды противоречивы), задвижка (клапан) не меняет своего положения. Команды не имеют друг перед другом приоритета, и поэтому одновременная инициация обеих команд ведет к их взаимоблокировке.

Если «Команда Открыть» равна «1» и «Команда Закрыть» равна «0», блок управления изменяет положение клапана (задвижки), увеличивая его с постоянной скоростью, до достижения одного из следующих условий:

1 – «Положение» клапана равно «100» (полностью открыта);

2 – «Команда Открыть» становится равной «0».

Если «Команда Закрыть» равна «1» и «Команда Открыть» равна «0», блок управления изменяет положение задвижки до достижения одного из следующих условий:

1 – «Положение» задвижки равно «0» (полностью закрыта);

2 – «Команда Закрыть» равна «0».

При достижении значений «Положение» величины «100» или «0» сигналы на выходе из блока в портах с именами «Открыта» или «Закрыта», соответственно, принимают значения «1» (логическая Истина), в противном случае они равны «0» (логическая Ложь), тем самым сигнализируя о том, что задвижка находится в промежуточном положении.

Войдите в субмодель «БУЗ» и добавьте на схему следующие блоки:

* 1 блок **«Сумматор»** (закладка **«Операторы»**);
* 1 блок **«Интегратор с ограничением»** (закладка **«Динамические»**);
* 2 блока **«Константа»** (закладка **«Источники»**);
* 2 блока **«Логические операции»** (закладка **«Логические»**).

После установки данных блоков на поле схемы, соедините их линиями связи между собой (Рисунок 78).

Затем произведите настройку установленных блоков.

Для блоков **«Константа»** установите следующие значения свойства **«Значение»** на вкладке **«Свойства»** в диалоговом окне: для первого блока установите значение **«100»**, для второго блока установите значение **«0»**.

Для блока «Сумматор» установите значение весовых множителей [1,-1]. Вместо сумматора можно установить и сравнивающее устройство.

Рисунок 78. Структурная схема БУЗ

Таким образом, на выходе из блока «Сумматор» будет значение «1», когда «Команда Открыть» равна «1» и «Команда Закрыть» равна «0». На выходе из блока «Сумматор» будет значение «-1», когда «Команда Открыть» равна «0» и «Команда Закрыть» равна «1». Во всех остальных случаях на выходе сумматора будет нулевое значение «0». Поскольку все сигналы векторные, то в каждой линии связи будет передаваться массив сигналов.

Для формирования сигналов «Открыта» и «Закрыта» используется логический блок «Логические операции» из закладки «Логические» (Рисунок 79)

Рисунок 79. Блок логичесике операции

|  |  |
| --- | --- |
| Рисунок 80. Редактирование блока «Логические операции» | Блок производит логическую операцию над векторным сигналом. В нашем случае мы проводим сравнение векторного сигнала с константой. Блок проверяет каждый из сигналов вектора, входящего в блок, и возвращает вектор, в котором значение «1» (логическая Истина) принимают сигналы, которые равны константе (скалярному сигналу на втором входном порте).  В настройках блока необходимо указать тип второго блока «Скаляр». |

Блок «Интегратор с ограничением» реализует расчет положения задвижек (моделирует двигатель с постоянным быстродействием) на основании поступающих на вход сигналов: если вход равен «-1», то уменьшает положение, если «+1» то увеличивает.

В нашем простейшем примере мы считаем, что положение изменяется линейно с постоянной скоростью, одинаковой для всех задвижек. Скорость изменения пропорциональна коэффициенту усиления интегратора. Для обработки вектора сигналов необходимо задать вектор значений для каждого параметра блока.

Например, если блок рассчитывает 4 сигнала, то необходимо ввести четыре коэффициента усиления для каждого из сигналов в векторе: [1,1,1,1]. Для одинаковых по величине значений можно использовать запись 4#1, которая эквивалентна вектору (массиву) [1,1,1,1].

Войдите в диалоговое окно редактирования свойств блока «Интегратор с ограничением» и установите следующие свойства блока:

|  |  |
| --- | --- |
| Коэффициенты усиления | Z\_Count#1 |
| Минимальное значение | Z\_Count#0 |
| Максимальное значение | Z\_Count#100 |
| Начальные условия | Z\_Count#0 |

Закройте окно редактирования свойств нажатием кнопки «Ок».

Запись Z\_Count#1 позволяет заполнить массив значений для вектора. В зависимости от величины «Z\_Count» формируется различный массив значений. Поскольку «Z\_Count» вычисляется путем обращения к базе данных, то он будет соответствовать количеству задвижек, существующих в базе данных. Таким образом, данный блок будет всегда корректно обрабатывать вектор сигналов при изменении количества задвижек в базе данных.

Рисунок 81. Редактирование блока «Интегратор с ограничением»

Если схема набрана правильно, то ее можно запустить на расчет и убедиться , что по всем линиям связи передается вектор сигналов, состоящий из двух элементов, в соответствии с количеством задвижек в базе данных (Рисунок 82).

А в базе данных для всех задвижек значение сигнала «Закрыта» устанавливается в истину (в соответствии с начальными условиями интегратора).

Рисунок 82. Проверка работоспособности схемы БУЗ

Сохраните созданную модель.

# Создание простого алгоритма управления

## Постановка задачи для создания алгоритмов

В процессе разработки систем управления сложных объектов часто приходится решать задачу разделения системы на составные части, т.е. реализуется подход, когда сложный алгоритм собирается из более простых алгоритмов, решающих различные локальные и строго определённые задачи. В данном упражнении мы разберем одну из таких задач и покажем возможности среды SimInTech для ее решения.

Выполняя шестое учебное упражнение, мы смоделировали типовой блок управления задвижкой. В данном учебном упражнении мы составим два алгоритма управления, которые будут формировать управляющие команды для первой и для второй задвижек и подавать их на созданный блок. Каждый алгоритм управления будет помещен в свой собственный блок, созданный на основе блока **«Субмодель»**. Таким образом система управления задвижками будет разделена на три части: общая для всех задвижек часть – блок управления задвижкой, две остальные части – блоки формирования управляющих команд для отдельных задвижек.

Приступим:

1. Откройте созданный на предыдущем занятии файл проекта «Схема автоматики 2.prt»;
2. Добавьте на схему два новых блока «Субмодель» из закладки «Субструктуры». Создайте подписи для добавленных блоков «Алгоритм управления задвижкой Z1» и «Алгоритм управления задвижкой Z2». После этого схема обретет вид, показанный на рисунке ниже (Рисунок 83).

Рисунок 83. Схема модели управления

Данные алгоритмы будут управлять двумя задвижками комплексной модели.

## Алгоритм управления первой задвижкой

Войдите в субмодель «Алгоритм управления задвижкой Z1» и поместите на схему блок «Чтение сигналов» из группы **Данные** (Рисунок 84). Данный блок позволяет получать сигналы из базы данных аналогично блоку «Чтение сигнала из списка».

Рисунок 84. Блок «Чтение сигналов»

Поместите на схему блок выход «Выход Алгоритма» (Рисунок 84). Данный блок позволяет не только записывать в базу данных значение ранее занесенных сигналов, но также и создавать новые сигналы.

После добавления на схему блоков, она должна выглядеть как Рисунок 85.

Рисунок 85. Схема с алгоритма с добавленными блоками

## Редактирование блоков «Чтение сигналов» и «Выход алгоритма»

Перед тем, как приступать к редактированию блоков, обратите внимание на их внешний вид: блоки выполнены в виде таблиц. Это позволяет создавать алгоритм управления, внешний вид которого приближен к внешнему виду функциональных планов, используемых при проектировании систем управления, что очень полезно при подготовке документации.

Вызовите диалоговое окно редактора двойным кликом левой кнопки мыши на блоке «Чтение сигналов». Откроется окно настройки чтения/записи, изображенное на рисунке ниже (Рисунок 86).

Рисунок 86. Окно редактирования блока чтения сигналов

В данном диалоговом окне «Редактор чтения\записи сигналов» можно настроить внешний вид таблицы и список сигналов, получаемых из базы данных при помощи данного блока. Окно можно поделить на несколько областей:

* 1 область – область содержащихся сигналов – в данной области построчно перечисляются все используемые блоком сигналы;
* 2 область – настройка параметров отображения таблицы – в данной области можно настроить внешний вид таблицы;
* 3 область – кнопки редактирования таблицы – кнопки позволяют добавлять и удалять строки из таблицы, менять их расположение в таблице, связывать конкретную строку с сигналом базы данных и пр.

В алгоритме для управления первой задвижкой мы будем использовать сигнал «Давление в промежуточном узле». Произведем заполнение блока **«Чтение сигналов»**:

1. Войдите в редактор блока;
2. Добавьте несколько новых строк. Выберите одну из добавленных строк;
3. Нажмите кнопку **«Заполнить выделенное из базы»**;
4. В появившемся окне «Редактор базы данных» выберите последовательно: Категория – «Датчики»; группа сигналов – «D1»; сигналы и данные для групп – «iq01» (Рисунок 87);
5. Нажмите кнопку «Добавить» в панели «Выбранные данные»;

Рисунок 87. Окно редактора базы данных

1. Закройте редактор базы данных нажав на кнопку «Ок».
2. Произведите настройку внешнего вида таблицы с помощью кнопок, приведя внешний вид таблицы к показанному на следующем рисунке (Рисунок 88);

Рисунок 88. Настройка блока чтения сигнала

1. Закройте окно **«Редактор чтения\записи сигналов»** нажатием кнопки **«Oк»**.

Аналогично произведите настройку блока «Выход алгоритма», выбрав из базы данных сигналы управления первой задвижкой: «Команда Открыть» и «Команда Закрыть». В столбце ID введите «B1» для строки «Команда Открыть» и «B2» для строки «Команда Закрыть». В поле «Имя алгоритма» введите строку «al01» (Рисунок 89).

Рисунок 89. Настройка блока «Выход алгоритма»

Если все настройки выполнены правильно, то схемное окно блока управления первой задвижкой должно содержать две таблицы, как представлено на рисунке (Рисунок 90):

Рисунок 90. Внешний вид схемы после настройки блоков «Чтение сигналов» и «Выход алгоритма»

## Структурная схема управления первой задвижкой

В первом варианте системы управления мы проводили управление задвижкой и расчет необходимого положения задвижки в одном алгоритме, и передавали вычисленное положение в теплогидравлическую модель. В текущем варианте мы будем управлять задвижкой в одном алгоритме, формируя две управляющие команды «Команда Открыть» и «Команда Закрыть», передавать их в следующий алгоритм (модель) – блок управления задвижкой. В БУЗ-е будет моделироваться и двигатель, и положение задвижки будет поступать в теплогидравлику.

Текущий алгоритм управления осуществляет сравнение сигнала «Давление в узле» с заданной константой, выполняющей роль задатчика давления. В случае, если давление выше заданного, то формируется команда на открытие задвижки, если же давление ниже заданного, формируется команда на закрытие задвижки. Структурная схема данного алгоритма представлена на рисунке ниже (Рисунок 91). Попробуйте самостоятельно набрать схему приведенного алгоритма, используя следующие блоки:

* 2 блока **«Константа»** (закладка **«Источники»**);
* 1 блок **«Сравнивающее устройство»** (закладка **«Операторы»**);
* 1 блок **«Операция БОЛЬШЕ»** (закладка **«Логические»**);
* 1 блока **«Операция МЕНЬШЕ»** (закладка **«Логические»**).

Рисунок 91. Внешний вид схемы после настройки блоков «Чтение сигналов» и «Выход алгоритма»

## Проверка работы алгоритма управления первой задвижкой

Для проверки работы алгоритма запустим созданную схему на расчет. Схема алгоритма управления примет вид, представленный на рисунке ниже (Рисунок 92):

Рисунок 92. Вид схемы алгоритма во время моделирования

Поскольку расчет производится без подключения теплогидравлической модели, значение сигнала «Давление в узле» в базе данных равно нулю. Сравнивающее устройство, произведя обработку сигнала, на выходе имеет значение «–117000», и блок **«Операция МЕНЬШЕ»** формирует значение «1» (логическая Истина) для сигнала «Команда Открыть».

Не останавливая расчет, перейдите в блок управления оборудованием и выполните двойной клик по линии связи между блоками «Команда Открыть» и «БУЗ» (Рисунок 93). В появившемся окне «Просмотр значений на линии связи» отражается список значений сигнала «Команда Открыть» для всех задвижек, занесенных в базу данных. Поскольку управление осуществляется только первой задвижкой, то и значение равное «1» имеет только первый элемент списка (Рисунок 93).

Рисунок 93. Вид схемы субмодели «Управление оборудованием» во время моделирования

Осуществите двойной клик на графике, в который передается положение задвижек из блока **«БУЗ»**. Созданная модель блока управления и двигателя задвижки осуществляет отработку сигнала «Команда Открыть» для первой задвижки и производит изменение положения задвижки с постоянной скоростью. При достижении положения «100» (полностью открыта) положение не изменяется; вторая задвижка не двигается и остаётся в положении «0» (полностью закрыта, Рисунок 94):

Рисунок 94. График изменения положения задвижки Z1 и Z2

## Алгоритм управления второй задвижкой

Для управления второй задвижкой мы создадим алгоритм, который будет менять положение задвижки по заданной последовательности программным способом.

Последовательность изменения будет задаваться блоком «Кусочно линейная» (зависимость) из закладки «Источники» (Рисунок 95):

Рисунок 95. Блок «Кусочно линейная» (зависимость) на закладке «Источники»

Алгоритм управления осуществляет сравнение текущего положения задвижки с заданным положением, полученным с помощью блока «Кусочно линейная» из закладки «Источники». В случае расхождения подается команда на открытие или закрытие задвижки.

Для снятия непрерывных включений и выключений задвижки используется релейное звено с зоной нечувствительности – блок «Релейное с зоной нечувствительности» из закладки «Нелинейные». Этот блок позволяет не посылать команду в том случае, когда отклонение положения находится в пределах заданной точности.

Рисунок 96. Блок «Релейное с зоной нечувствительности»

Схема алгоритма управления второй задвижкой приведена на следующем рисунке (Рисунок 97). В схеме данного алгоритма используются такие же блоки «Чтение сигналов» и «Выход алгоритма», какие мы использовали в алгоритме управления первой задвижкой. **Примечание**: для ускорения набора схемы, т.к. второй алгоритм во многом похож на первый, можно не набирать его «с нуля», а скопировать всю первую субмодель в буфер обмена, удалить вторую субмодель и осуществить вставку из буфера копии первой, затем её отредактировать.

Рисунок 97. Схема управления второй задвижкой

Войдите в субмодель «Алгоритм управления задвижкой Z2» и соберите схему, как показано на рисунке (Рисунок 97). Для блока «Чтение сигналов» задайте параметры, как показано на следующем рисунке (Рисунок 98):

Рисунок 98. Параметры блока чтение сигналов

Для блока «Выход алгоритма» задайте параметры, как показано на следующем рисунке (Рисунок 99).

Рисунок 99. Параметры блока выход алгоритма

Для блока «Релейное с зоной нечувствительности» задайте параметры, как показано на следующем рисунке (Рисунок 100):

Рисунок 100. Параметры блока релейное с зоной нечувствительности

Работает данное звено следующий образом: выход блока y(t) либо принимает одно из трех значений Y1, 0, Y2 либо не изменяется: y(t) = y(t – Δt), где y(t – Δt) – значение выхода на предыдущем шаге интегрирования (на предыдущем шаге численного расчета).

y(t) = Y1, если x(t) < a1 – реле переключается на нижнее значение при уменьшении входного воздействия ниже нижней границы переключения.

y(t) = Y2, если x(t) > b1 – реле переключается на верхнее значение при увеличении входного воздействия выше верхней границы переключения.

y(t) = 0, если a < x(t) < b – реле находится в нулевом положении если, значение входа попадает в зону нечувствительности.

y(t) = y(t-Δt), если a1 ≤ x (t) ≤ a или b ≤ x (t) ≤ b2 – значение реле не изменяется, входное воздействие не пересекает зону переключения.

Заданные параметры блока (Рисунок 100) позволяют регулировать, или управлять положением задвижки с точностью ±1% от заданного значения.

Для задатчика положения второй задвижки используется кусочно-линейная зависимость от времени. Блок «Кусочно линейная» позволяет задать массив значений функции в различные моменты времени. Между заданными точками происходит линейное изменение значения выхода блока.

|  |  |
| --- | --- |
| Рисунок 101. Параметры блока «Кусочно линейная» | Зададим алгоритм для блока «Кусочно линейная»:   * в промежуток времени 0 – 100 секунд значение 10; * в промежутке 100 – 200 секунд значение возрастает до 40. * в промежутке 200 – 300 секунд значение 40; * в промежутке 300 – 400 значение убывает до 20; * в промежутке 400 – 500 (и далее) значение 20.   Для реализации этого алгоритма задайте параметры блока «Кусочно линейная» как показано на рисунке (Рисунок 101). |

## Проверка работы модели

Для детальной проверки работы алгоритма можно замедлить выполнение процесса моделирования. Для этого нажмите кнопку «Параметры расчета» на схемном окне (Рисунок 102)

Рисунок 102. Кнопка вызова параметров расчета

В диалоговом окне «Свойства решателя» перейдите на закладку «Синхронизация» и установите галочку «Синхронизировать с реальным временем». Задайте коэффициент ускорения равным **0.5** (Рисунок 103).

Рисунок 103. Настройка скорости расчета в два раза медленнее реального времени

Перейдите не закладку «Параметры расчёта» и установите конечное время расчета равным 500 секунд (Рисунок 104).

Рисунок 104. Настройка конечного времени расчета

Запустите созданную схему на расчет.

Остановите расчет через 5 – 10 секунд после начала, нажав кнопку «Пауза» в главном окне программы.

Перейдите в субмодель «Управление оборудованием» и осуществите двойной клик по лини связи между блоками «Команда Открыть» и «БУЗ» (Рисунок 93). В появившемся окне «Просмотр значений на линии связи» отражается список значений сигнала «Команда Открыть» для всех задвижек, занесенных в базу данных. Поскольку мы добавили алгоритм управления второй задвижкой в начальный момент времени, значение, равное «1», имеют оба элемента векторного сигнала (Рисунок 105).

Рисунок 105. Вид схемы субмодели «Управление оборудованием» во время моделирования

Откройте «Редактор базы данных», установите «Режим просмотра значений» и **«Обновлять с интервалом»** и убедитесь, что значения сигналов для задвижек в базе данных соответствуют значениям, рассчитанным в блоке управления задвижками (Рисунок 106). Для отображения такого сводного режима просмотра надо выделить необходимые группы сигналов (с зажатой клавишей Shift для выделения диапазона групп сигналов или с зажатой клавишей Ctrl для выделения индивидуальных групп сигналов) и перейти в закладку «**Сводная**». Также можно воспользоваться кнопкой в интерфейсе редактора базы сигналов, которая выделяет все группы сигналов в выбранной категории.

Рисунок 106. Значение сигналов в базе данных

При создании алгоритма с использованием блока «Выход алгоритма» в зависимости от его настроек (в общих свойствах если заполнено свойство «**Имя категории для алгоритмов**») может происходить автоматическое добавление в базу данных новых алгоритмов, чтобы при разработке системы управления можно было использовать выходы алгоритмов в качестве входных воздействий в других частях модели системы управления. В нашем случае (для версии SimInTech-1.3.1.25 от 4 декабря 2015) в базе данных не появилась никакая новая категория, т.к. пол умолчанию указанное свойство равно пустой строке. Однако в других проектах эти блоки могут добавлять сигналы в базу автоматически.

Продолжите расчет модели (нажатием на кнопку «Пуск» главной панели управления).

|  |  |
| --- | --- |
| Рисунок 107. Положение задвижек при автономном моделировании схемы автоматики 2 | Перейдите в субмодель «БУЗ», осуществите двойной щелчок по графику положения задвижек и убедитесь, что график примерно соответствует изображенному на рисунке ниже (Рисунок 107). Для ускорения расчета можно непосредственно в процессе счета поменять коэффициент ускорения расчета.  Первая задвижка открывается полностью. Вторая задвижка изменяет свое положение согласно кучосно-линейной зависимости, установленной в алгоритме управления. Поскольку в алгоритме управления второй задвижкой присутствует блок «Релейное с зоной нечувствительности» изменение положения происходит «ступеньками», в моменты когда отличие текущего положения от заданного превышает 1%. |

В целях отладки иногда бывает удобно или нужно быстро вывести на график две или более переменных для взаимного сравнения. В качестве самостоятельного упражнения, выведите на график кроме текущих положений задвижек ещё и заданное положение для второй задвижки и убедитесь, что перемещение второй задвижки осуществляется только в те моменты, когда её положение начинает отличаться от заданного на 1%, а прекращает своё движение задвижка только тогда когда отличие уменьшается до величины менее 0,5%, как это и было задано в релейном блоке с зоной нечувствительности. Для подсказки приведены Рисунок 108 и Рисунок 109.

Рисунок 108. Вывод заданного положения Z2 на график

**Примечание**: для детализированного исследования в данном случае потребуется уменьшить шаг расчета (интегрирования) с заданного по умолчанию 0,25 с до порядка 0,1…0,01 секунды. Это настраивается в параметрах расчета схемы автоматики (Рисунок 104).

С шагом 0,25 точность расчета будет недостаточна, чтобы корректно отследить работоспособность алгоритма на уровне десятых долей процента по положению задвижки, т.к. в интеграторе (в модели БУЗ) задан коэффициент усиления **1**, и за один шаг расчета при шаге расчета 0,25 секунды задвижка перемещается на 0,25%. Соответственно, для того чтобы просмотреть работу релейного звена с зоной нечувствительности 1% и зоной возврата 0,5%, шаг расчета должен быть хотя бы на порядок меньше, т.е. 0,05 с а лучше 0,01 с.

Таким образом, в режиме автономного моделирования мы убедились, что реализованный блок управления задвижками корректно обрабатывает все (две) задвижки проекта, регулятор первой задвижки и программное управление второй задвижкой работают корректно, на заданном интервале 500 секунд.

Рисунок 109. Сравнительный анализ заданного положения Z2 и текущего

# Изменение комплексной модели

Использование базы данных сигналов в SimInTech позволяет легко формировать сложные модели из нескольких созданных ранее проектов. Главное условие – использование единой базы данных для обмена значениями сигналов. Зачастую в сложных проектах используется несколько пакетов проектов, где в зависимости от задачи моделирования/расчета подключаются не все расчетные модели, а только требуемые для данного режима.

|  |  |
| --- | --- |
| Рисунок . Окно управления пакетом проектов, удаление проекта из пакета | Созданная при выполнении учебного задания 5 комплексная модель может быть легко изменена путем замены одной «системы управления» на другую. В данном учебном задании мы протестируем модель системы управления из учебного задания 7 совместно с теплогидравлической моделью из учебного задания 3. Для этого заменим в пакете схему автоматики 1 на схему автоматики 2.  Откройте файл пакета комплексной модели управления «Pack1.pak» из учебного задания 5. В окне управления пакетом выделите проект «Схема автоматики 1.prt» и нажмите кнопку «Удалить проект» (Рисунок 110). |

Нажмите кнопку «Добавить проект» и выберите в стандартном меню открытия файла проект модели, созданный при выполнении учебного задания 7 – «Схема автоматики 2.prt». Сохраните пакет проектов либо со старым именем, или под новым именем, например, «**Pack2.pak**».

## Проверка изменённой комплексной модели

Запустите комплексную модель на расчет, используя кнопку «Пуск» в окне управления «Пакет». Если создание проектов выполнено без ошибок, то должен начаться расчет комплексной модели. Осуществите клик на схеме гидравлической модели.

Надписи задвижек должны отображать их положение, полученное из системы управления. Например, на 35–ой секунде расчета вторая задвижка находится в положении ~ 9,7%, а первая продолжает открываться-прикрываться согласно алгоритму и держится в районе ~ 7%. Постоянное перемещение вперёд-назад первой задвижки свидетельствует, мягко говоря, о явном несовершенстве предложенного алгоритма регулирования и поддержания давления…

Рисунок 111. Схема гидравлической системы на 35 секунде расчета

После 100–й секунды расчета второй клапан начинает открываться согласно алгоритму управления, при этом для поддержания давления в узле первый клапан так же начинает открываться.

После 200–й секунды расчета «Z2» занимает положение 40%, а первый – 52,8%.

После 400–й секунды расчета вторая задвижка занимает положение 20.8%, а первая – 25,8%.

Перейдите в субмодель «Управление оборудованием». Осуществите двойной клик на блоке графиков положения задвижек. По окончании расчета график должен иметь вид, представленный на рисунке (Рисунок 112).

Из графика видно, что первый клапан, который работает на поддержание давления согласно алгоритму управления, непрерывно находится «в работе», осуществляя колебательные движения, это говорит о том, что созданный алгоритм управления не совсем удачный для данных условий работы системы.

Для корректировки регулятора давления можно добавить в алгоритм управления задвижкой Z1 релейный блок, аналогичный блоку, используемому в алгоритме управления задвижкой Z2.

Рисунок 112. Положение задвижек в режиме расчета комплексной модели

Перейдите в теплогидравлическую модель и откройте график давления во внутреннем узле. Увеличите масштаб графика, как показано на следующем рисунке (Рисунок 113). Анализируя график давления, можно сделать вывод о необходимой ширине «зоны нечувствительности» релейного блока для «Алгоритма управления задвижкой Z1».

Рисунок 113. График давления во внутреннем узле

## Самостоятельная работа

1. Измените алгоритм управления задвижкой Z1 таким образом, чтобы поддержание давления на уровне **117 кПа** во внутреннем узле не приводило бы к постоянному перемещению задвижки Z1. График положения задвижек должен выглядеть примерно так, как показано на рисунке (Рисунок 114). Изменение давления и величины уставок по давлению (зона нечувствительности 2 кПа, зона возврата -1,5 кПа) приведены на следующем рисунке.

По приведённым графикам видно, что скорректированный регулятор гораздо реже «дёргает» двигатель задвижки и переходной процесс идёт более «гладко».

Рисунок 114. График положения задвижек после корректировки алгоритма регулирования Z1

Рисунок . График давления в среднем узле, и «уставок» регулятора

Также, обратите внимание на начальный этап моделирования – из-за неравновесного начального состояния, заданного в модели, в первые секунды моделирования происходит его установление и из-за этого скачки давления существенны. Для устранения этого эффекта необходимо либо более корректно выставить начальное состояние в модели, либо воспользоваться механизмом рестартов (его описание приведено в справочной системе и не входит в настоящие упражнения).

1. Факультативно: создайте новый проект (третий) системы управления, в котором алгоритм управления задвижкой Z1 поддерживает расход через систему на уровне 20. Для этого потребуется добавить точку контроля по расходу через какой-либо из каналов, и скорректировать регулятор.

# Создание окна управления оборудованием

## Ручное управление в проекте

SimInTech является открытой программной средой, позволяющей при математическом моделировании использовать данные, полученные из различных источников. Одним из возможных источников являются окна управления, в которых пользователь может задавать различные воздействия на математическую модель, используя интерактивные элементы, такие как виртуальные кнопки, переключатели, ручки и т.п.

В качестве примера окна управления мы сделаем окно управления задвижкой для схемы теплогидравлики, созданной в предыдущих учебных заданиях.

## Создание окна управления

1. Откройте файл с гидравлической моделью **«Схема теплогидравлики 1.prt»**;
2. В главном окне SimInTech нажмите кнопку «Менеджер данных» (Рисунок 116);



Рисунок 116. Кнопка вызова менеджера данных

Нажатие на данную кнопку вызывает на экран диалоговое окно «Менеджер данных» (Рисунок 117). Данное окно служит для настройки различных инструментов пост-обработки данных, настройки обмена данными с другим ПО или файлами, а также создания каналов воздействия на математическую модель (панелей управления оборудованием). Окно состоит из двух областей:

* 1 область – область кнопок;
* 2 область – область дерева отображаемых объектов.



Рисунок 117. Диалоговое окно «Менеджер данных»

1. Нажмите кнопку **«Добавить категорию»** (Рисунок 117). Введите название новой категории **«Окна управления оборудованием»**. Переименование категории, как и любого другого объекта **«Менеджера данных»**, можно осуществить нажатием правой кнопки мыши на этом объекте и выбрав в выпадающем меню пункт **«Переименовать»**.
2. Выделите созданную категорию и нажмите кнопку **«Окно анимации»** (Рисунок 118):



Рисунок 118. Окно «Менеджер данных» после добавления новой категории

В категории «Окна управления оборудованием» появится новый элемент «Окно анимации». При необходимости раскройте список категорий, нажав на значок «+» слева от имени категории.

1. Введите имя для вновь созданного элемента «Окно управления задвижкой» (Рисунок 119):



Рисунок 119. Окно «Менеджер данных» после добавления новой категории

1. Осуществите двойной клик мыши на «Окне управления задвижкой».

После этого откроется окно редактора (Рисунок 120), в котором будет происходить создание панели управления и панель примитивов (Рисунок 121), из которых будут формироваться элементы управления оборудованием. Созданное в менеджере данных окно анимации доступно и может быть вызвано специальным механизмом ссылки (Instance) из любой части (из любого блока) математической модели в процессе моделирования.



Рисунок 120. Пустое окно управления задвижкой



Рисунок 121. Панель примитивов

## Создание интерфейса управления оборудованием

Используя набор примитивов, пользователь может собрать и настроить внешний вид окна управления оборудованием на основе окна анимации. Окно анимации может содержать как элементы отображения, которые меняют свой внешний вид в зависимости от какого-либо внешнего события, так и интерактивные элементы управления, которые позволяют воздействовать на сигналы в базе данных математической модели.

Окно управления задвижкой в нашем примере будет содержать две кнопки, одна из которых позволяет послать команду на открытие задвижки, другая – на закрытие.

Для выбора примитива следует осуществить одиночный клик левой мышкой на соответствующей кнопке панели примитивов, затем осуществить одиночный клик в окне управления в том месте, куда желательно поместить примитив.

Разместите на окне управления задвижкой следующие элементы панели примитивов:

* «Кнопка» – два элемента;
* «Текст» – три элемента;
* «Линейный прибор» – один элемент.

Расположите примитивы относительно друг друга приблизительно так, как показано на следующем рисунке (Рисунок 122):



Рисунок 122. Окно управления задвижкой

Каждый примитив представляет собой объект, свойства которого можно редактировать. Для редактирования свойств примитива необходимо выполнить следующие действия:

1. Выделить объект, осуществить клик правой кнопкой мыши, в всплывающем окне выбрать пункт меню «Свойства объекта» (Рисунок 123);



Рисунок 123. Всплывающее меню редактирования примитива

1. После этого появится диалоговое окно редактирования свойств объекта, в котором пользователь может изменить свойства выбранного примитива (Рисунок 124).



Рисунок 124. Окно редактирования примитива

Измените свойства примитивов следующим образом:

1. Задайте в свойстве **«Имя объекта»** (верхняя строчка диалогового окна, Рисунок 124) следующие имена:

* для кнопок – **Open\_Button** и **Close\_Button**;
* Для верхней текстовой подписи – **Name\_TextLabel**;
* Для линейного прибора – **Position\_Bar;**

1. Для улучшения внешнего вида окна управления задвижкой увеличьте размер шрифтов до **«15»** для текстовых надписей.

## Создание переменных окна управления задвижкой

Для корректной работы окна управления необходимо осуществить программирование действий пользователя с примитивами в сигналы для математической модели. В первую очередь окно управления оборудованием должно получить имя объекта математической модели, для которого это окно вызвано.

Перейдите в главное меню «Окна управления задвижкой» и выберите пункт меню «Сервис» подпункт «Глобальные свойства…» (Рисунок 125). В появившемся диалоговом окне «Общие свойства» (Рисунок 126) необходимо добавить новое свойство для панели управления.

Внимание!!! Если имя добавляемого сигнала в окне управления совпадает с именем свойства объекта, для которого вызвано данное окно, то его значение автоматически устанавливается равным значению свойства объекта.

Например, в данном случае мы добавим сигнал «Name», при вызове данного окна его значение станет равным имени задвижки.



Рисунок 125. Вызов окна добавления свойств



Рисунок 126. Окно добавления общих свойств

Нажмите кнопку добавить сигнал и введите следующие значения (Рисунок 126):

|  |  |
| --- | --- |
| Имя | Name |
| Название | Имя объекта |
| Режим | Вход |
| Тип данных | Строка |

Для отображения состояния задвижки мы будем использовать переменную, которая присутствует в свойствах объекта «Задвижки общего вида» кода HS – «Состояние»:

1. Добавьте новый сигнал и настройте его свойства в соответствии с тем, что показано на следующем рисунке (Рисунок 127):



Рисунок 127. Окно добавления общих свойств

1. Закройте диалоговое окно нажатием кнопки «Ок».

Таким образом мы добавили две переменные, в которые будут передаваться имя и положение задвижки, для которой мы вызвали окно управления во время моделирования.

1. Перейдите в главное меню **«Окна управления задвижкой»** и выберите пункт меню **«Сервис»**, подпункт **«Локальные переменны…»** (Рисунок 125). В появившемся диалоговом окне **«Локальные переменные»** (Рисунок 128) необходимо добавить те сигналы, которые будет отображать (и формировать) данное окно управления. В нашем случае мы будет отображать **«Положение»** задвижки и посылать команды **«Команда Открыть»** и **«Команда Закрыть»**.

В нашем случае для управления задвижкой будут использоваться сигналы, созданные при формировании базы данных во время выполнения предыдущих учебных заданий. Напомним правила формирования сигналов в базе данных:

Сигнал в базе данных состоит из имени объекта и имени сигнала, разделенных нижним подчеркиванием. Например, сигнал «Положение» задвижки «Z1» в базе данных имеет имя «Z1\_xq01».

Для формирования правильного имении сигнала управления следует в локальных переменных создавать имена, используя знак нижнего подчеркивания в начале имени (Рисунок 128).



Рисунок 128. Окно локальных переменных

1. Добавьте локальные переменные как показано на рисунке выше (Рисунок 128) и закройте окно нажатием кнопки **«Ок»**.

## Программирование окна управления задвижкой

Созданные ранее элементы интерфейса и набор локальных и общих переменных позволяют осуществить программирование окна управления задвижкой таким образом, чтобы во время моделирования осуществлять «вручную» воздействие на математическую модель. Для завершения создания панели управления необходимо запрограммировать логику поведения отображающих и управляющих элементов интерфейса.

Перейдите в главное меню «Окна управления задвижкой» и выберите пункт меню «Сервис» подпункт «Скрипт…» (Рисунок 129).



Рисунок 129. Вызов окна программирования

В появившемся текстовом окне «Язык программирования» введите следующий текст программы, как показано ниже (Рисунок 130):



Рисунок 130. Скрипт управления задвижкой

Вся программа заключена между ключевыми словами «formattext» и «end;». Использование данного ключевого слова **«formattext»** позволяет формировать имя переменной, используя шаблон. Вместо выражения, заключенного в фигурные скобки, будет подставлено его значение. В нашем случае при вызове диалогового окна значению общей переменной «Name» будет присвоено значение данного свойства у задвижки: для задвижки «Z1» после вызова окна управления выражение типа {Name}\_yb02 будет преобразовано в соответствии с шаблоном в выражение Z1\_yb02, которое совпадает с именем сигнала в базе данных.

Строки, заключенные между ключевыми словами «initialization» и «end;», выполняются один раз при открытии окна:

* Name\_TextLabel.Text = "Положение задвижки " + Name; –

присвоение тексту верхней надписи строки с именем задвижки для которой вызвано окно управления.

* Position\_Bar.Value = State; – отображение линейным прибором степени открытия задвижки.
* Open\_Button.Down = {Name}\_yb01; – приведение внешнего вида кнопки «Открыть» в соответствие с сигналом базы данных.
* Close\_Button.Down = {Name}\_yb02; – приведение внешнего вида кнопки «Закрыть» в соответствие с сигналом в базе данных.

Строки основного текста программы выполняются на каждом шаге моделирования, пока окно управления активно:

* {Name}\_yb01 = Open\_Button.Down; – отправка в базу данных сигнала команды открытия задвижки.
* {Name}\_yb02 = Close\_Button.Down; – отправка в базу данных сигнала команды закрытия задвижки.
* Position\_Bar.Value = State; – отображение линейным прибором степени открытия задвижки.

Закройте окно нажатием кнопки «Применить» в левом верхнем углу, сохранив внесенные изменения.

## Связь задвижки с окном управления

Для подключения окна управления к объекту на схеме необходимо выполнить следующие действия:

1. Перейдите в Схемное окно теплогидравлической модели;
2. Выделите задвижку «Z1» и вызовите окно редактирования свойств (Рисунок 131);



Рисунок 131. Окно редактирования свойств задвижки Z1

1. В строке «Ссылка» нажмите кнопку редактирования. Откроется диалоговое окно выбора ссылки.



Рисунок 132. Окно редактирование задвижки

1. В появившемся диалоговом окне выберите категорию «Панели управления» и элемент «Окно управления задвижкой» (Рисунок 132).
2. В строке «Режим показа формы» выберите «Управление объектом» (Рисунок 132). Нажмите кнопку **«Ок»**;
3. Выйдете из окна редактирования свойств нажатием на кнопку **«Ок»**;
4. Аналогично установите связь с окном управления для второй задвижки **«Z2»**;
5. Перейдите на теплогидравлическую схему, установите режим «Индикация», используя кнопку в верхней части окна (Рисунок 133):



Рисунок 133. Схемное окно теплогидравлической модели

1. Запустите модель на расчет. Убедитесь, что при двойном клике на задвижке появляется окно управления задвижкой;
2. Убедитесь, что при нажатии кнопок на панели управления задвижкой, значения сигналов в базе данных **«Команда Открыть»** и **«Команда Закрыть»** меняются;
3. Сохраните проект **«Схема теплогидравлики 1.prt»**;
4. Закройте проект.

## Ручное управление задвижкой в комплексной модели

1. Откройте комплексную модель «pack1.pak» созданную при выполнении учебного задания 8.

В данную комплексную модель входят два проекта:

* **«Схема теплогидравлики 1.prt»** – теплогидравлическая модель;
* «Схема автоматики 2.prt» – модель системы управления.

Обе этих модели загружаются автоматически при загрузке пакета;

1. Убедитесь, что теплогидравлическая модель содержит ранее созданное окно управления. Для этого в главном окне SimInTech нажмите кнопку «Менеджер данных» (Рисунок 116) и проверьте наличие добавленной категории с окном.

Поскольку «Схема автоматики 2.prt» осуществляет постоянно управление задвижками, для исключения взаимного влияния автоматического управления и ручного управления друг на друга мы осуществим отключение алгоритма управления задвижкой «Z2»;

1. Для этого необходимо перейти в модель автоматики и выделить субмодель «Алгоритм управления задвижкой Z2».



Рисунок 134. Схемное окно модели автоматики

1. Перейдите в главное окно программы, в котором выберите пункт меню «Правка», подпункт «Исключить объекты» (Рисунок 135). Блоки, исключенные из расчета, на схеме отображаются черным цветом и при моделировании не участвуют в обмене сигналами.

Таким образом, мы отключили в схеме алгоритм управления второй задвижкой и исключили ситуацию, когда сигналы, настроенные пользователем через «Окно управления задвижкой», противоречат сигналам из системы управления;



Рисунок 135. Меню исключения блока

1. Сохраните проект «Схема автоматики 2.prt»;
2. Запустите комплексную модель на расчет;
3. Осуществите на теплогидравлической схеме двойной клик на второй задвижке;
4. В появившемся окне управления подайте команды на открытие и закрытие задвижки. Убедитесь, что математическая модель корректно отрабатывает сигналы на открытие и закрытие (Рисунок 136).



Рисунок 136. Управление второй задвижкой в «ручном режиме»

# Создание журнала регистрации событий

## Регистрация событий

При работе со сложной математической моделью требуется анализировать множество параметров, как аналоговых, так и дискретных. SimInTech позволяет отслеживать параметры системы в виде временных графиков, фазовых портретов, текстовых таблиц и виртуальных приборов, макетов панелей (щитов) управления. Также существенную помощь в анализе оказывает журнал регистрации событий, который позволяет осуществить запись последовательности любых событий в математической модели. Анализ этих записей позволяет восстанавливать последовательность событий.

В SimInTech существует инструмент регистрации событий, который позволяет создавать один или несколько журналов событий для всей математической модели или любой ее части.

## Создание журнала регистрации событий

Откройте файл с гидравлической моделью **«Схема теплогидравлики 1.prt»**. созданный при выполнении предыдущих учебных заданий. Приступим к созданию журнала регистрации событий:

1. В главном окне программы нажмите кнопку «Менеджер данных» (Рисунок 137):



Рисунок 137. Кнопка вызова менеджера данных

1. Откроется диалоговое окно «Менеджер данных» (Рисунок 138);



Рисунок 138. Диалоговое окно «Менеджер данных»

1. Нажмите кнопку «Добавить категорию» (Рисунок 138). Введите название новой категории «Журналы регистрации событий»;
2. Выделите созданную категорию и нажмите кнопку «Журнал событий» (Рисунок 139);



Рисунок 139. Диалоговое окно «Менеджер данных» после добавления новой категории

1. В категории «Журналы регистрации событий» появится новый элемент «Регистратор событий» (Рисунок 140). При необходимости раскройте список категорий, нажав на значок «+» слева от имени категории:



Рисунок 140. Диалоговое окно «Менеджер данных» после добавления «Регистратора событий»

Событием в математической модели является любое изменение расчетного параметра. Для создания события нужно выбрать параметр, изменение которого будет являться событием, и настроить его свойства.

## Добавление параметров в «Регистратора событий»

Для создания нового события необходимо осуществить следующие действия:

1. Осуществите клик правой кнопкой мыши на пункте «Регистратор событий».
2. В выпадающем меню выберите пункт **«Добавить параметр»** (Рисунок 141):



Рисунок 141. Добавление параметра в журнал регистрации событий

1. В появившемся диалоговом окне введите имя блока «Z1» (первая задвижка) и имя параметра «State» (положение задвижки) (Рисунок 142):



Рисунок 142. Изменение параметра для регистрации

1. Закройте окно нажатием на кнопку **«Ок»**, сохранив внесенные изменения.

В окне «Менеджер данных» под пунктом «Регистратор событий» появится новый параметре «Z1.Sate».

Кроме добавления нового параметра по имени блока можно добавлять в качестве параметров сигналы из базы данных проекта. Для этого:

1. повторите приведенные выше пункты 1–2, в диалоговом окне «Изменение параметра» нажмите кнопку «Найти значение в базе» (Рисунок 143);



Рисунок 143. Кнопка вызова поиска параметров в базе данных

Нажатие кнопки «Найти значение в базе данных» приводит к появлению диалогового окна редактора базы данных.



Рисунок 144. Выбор сигналов из базы данных

1. Выберите в базе данных сигнал **«yb01»** («Команда Открыть» для задвижки «Z2»;
2. Аналогичным образом добавьте параметр «yb02» («Команда Закрыть») для задвижки **«Z2»**.

## Настройка параметров регистрации событий

Кроме выбора параметра математической модели для регистрации события необходимо выбрать условия возникновения события. Например, событием может быть превышение значения выше уставки (т.е. определенной величины, с которой сравнивается текущий расчетный сигнал).

По умолчанию регистратор настроен на изменение значения логических параметров с «0» (логическое «Нет») на «1» (логическое «Да»).

Для изменения условий срабатывания события необходимо выполнить следующие действия:

1. Осуществите клик правой кнопкой мыши по названию параметра в разделе «Регистратор событий»;
2. В выпадающем меню выберите пункт «Дополнительно»:



Рисунок 145. Вызов диалогового окна настройки события

После этого появляется окно настроек регистрации событий (Рисунок 146). В данном диалоговом окне необходимо настроить следующие параметры:

— Режим регистрации – определяет изменение параметра, которое приводит к появлению события. Возможные варианты:

* Увеличение значения;
* Уменьшение значения;
* Изменение значение;
* Превышение уставки;
* Снижение ниже уставки;

— Приоритет – определяет очередность регистрации событий в журнале, для событий, которые произошли одновременно, первым записывается событие с более высоким приоритетом;

— Уставка – числовая величина, с которой происходит сравнение значения параметра;

— Описание события – текст сообщения о событии, который записывается в журнал событий;



Рисунок 146. Настройка параметров регистрации событий

1. Задайте для параметра «Z1.State» следующие значения:

|  |  |
| --- | --- |
| Режим регистрации | Превышение уставки |
| Приоритет | 0 |
| Уставка | 99.99 |
| Описание события | Задвижка Z1 открыта полностью |

1. Нажмите на кнопку **«Ок»**, сохранив внесенные изменения.

Для событий, связанных с параметрами **«Команда Открыть»** и «Команда Закрыть» для задвижки «Z2», параметры регистрации событий настроены по умолчанию, так что появление команд автоматически приводит к появлению событий.

1. Сохраните проект **«Схема теплогидравлики 1.prt»**.

## Окно «Регистратор событий»

Для вызова окна «Регистратор событий» необходимо осуществить двойной клик на соответствующем пункте в окне «Менеджер данных». При этом появится окно аналогичное изображенному на рисунке ниже (Рисунок 147). Данное окно содержит в себе две закладки:

* «Журнал» – содержит таблицу, в которую выводится список событий математической модели, а также панель с кнопками управления данным списком;
* «Настройки» – предназначены для настройки «Регистратора событий».

Панель управления на закладке «Журнал» содержит следующие кнопки:

* «Поверх всех окон» – включает и выключает этот режим для «Регистратора событий»;
* «Очистить» – удаляет все существующие записи в «Регистраторе событий»;
* «Открыть» – позволяет загрузить сохраненный ранее список событий;
* «Сохранить» – позволяет сохранить список событий в текстовый файл;
* «Удалить» – служит для удаления выбранного события из списка сигналов;
* «Копировать» – позволяет скопировать существующий список событий в буфер обмена Windows.



Рисунок 147. Окно «Регистратор событий»

Таблица на закладке «Журнал» содержит следующие столбцы:

* «Время» – расчетное время математической модели, когда произошло событие;
* «Сигнал» – имя параметра, в формате внутреннего языка программирования, для которого регистрируются события;
* «Описание события» – текстовый строка, заданная при настройке события, либо для сигналов из базы данных данная строка соответствует значению в поле **«Название»** редактора базы данных;
* «Пред. значение» – значения параметра до события;
* «Новое значение» – значения параметра после события;
* «Приоритет» – значение приоритета.



Рисунок 148. Окно «Регистратор событий», закладка «Настройки»

Закладка «Настройки» содержит следующие элементы управления:

* Флаг «Записывать события в общее окно сообщений проекта» – при установке данного флага все сообщения в «Регистраторе событий» дублируются в окне сообщений проекта;
* Флаг «Записывать события в окно журнала» – при установке этого флага сообщения о событиях не выводятся в окно журнала;
* Поле «Имя файла лога событий» – позволяет задать имя файла, в который будут сохраняться сообщения о событиях;
* Флаг «Добавлять события в существующий файл» – при установке этого флага существующие записи в файле не стираются;
* Поле «Файл списка сигналов» – позволяет задать файл, в котором хранится список параметров, на основании которых создается журнал регистрации событий;
* Кнопки рядом предназначены для:
* открытия диалогового окна выбора файла;
* обновления списка сигналов из файла;
* сохранение списка сигналов в файл;
* Строка «Формат списка сигналов» – позволяет задать параметры, и их последовательность, в которой они будут записываться в текстовый файл списка сигналов;
* Выпадающее меню «Формат вывода модельного времени» – позволяет задать формат вывода модельного времени:
* «Десятичный» – время выводится в секундах;
* «Часы: минуты: секунды» – время выводится с использование часов минут и секунд.

## Использование журнала регистрации событий при моделировании

Воспользуемся журналом регистрации событий. Для этого выполните следующие действия:

1. Откройте комплексную модель «pack1.pak», созданную при выполнении учебного задания 9;
2. Перейдите в окно проекта «Схема теплогидравлики 1.prt». Для этого можно воспользоваться главным меню SimInTech, пункт – «Окно» (Рисунок 149);



Рисунок 149. Переключение между окнами комплексной модели

1. Убедитесь, что теплогидравлическая модель содержит ранее созданный журнал регистрации событий, для этого в главном окне программы нажмите кнопку «Менеджер данных»;
2. Осуществите двойной клик на пункте «Регистратор событий». Появится окно, в котором будет отображаться список событий, зарегистрированных в процессе моделирования (Рисунок 150).



Рисунок 150. Окно «Регистратор событий»

1. Для удобства просмотра событий можно установить режим «Поверх всех окон», нажав на соответствующую кнопку;
2. Запустите комплексную модель на расчет;
3. Осуществите двойной клик на второй задвижке в теплогидравлической модели. Откроется окно управления задвижкой;
4. В появившемся окне управления подайте команды на открытие и закрытие задвижки;
5. Убедитесь, что команды на открытие и закрытие задвижки регистрируются в журнале событий.



Рисунок 151. Регистрация событий в комплексной модели

1. Переведите задвижку **«Z2»** в полностью открытое состояние. Дождитесь, когда алгоритм управления задвижкой **«Z1»** выполнит полное открытие задвижки. Убедитесь, что факт полного открытия задвижки регистрируется журналом событий (Рисунок 151).

На этом учебные задания завершены. В данных упражнениях вы научились создавать и отлаживать автономно проекты автоматики и теплогидравлики (в базовом варианте), комбинировать их и создавать комплексные модели, объединённые общей базой сигналов, строить необходимые графики, настраивать шаг расчета, синхронизацию с реальным временем, создавать панели управления для типовых органов управления, познакомились с некоторыми расширенными инструментами SimInTech (регистратор событий, запросы к базе сигналов, векторная обработка и другие) и приобрели базовые навыки работы в среде SimInTech. Спасибо!