Министерство образования и науки РФ

Федеральное государственное бюджетное образовательное учреждение
высшего образования
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ ЛЕСОТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ имени С.М.Кирова

Кафедра информационных систем и технологий

РАСПРЕДЕЛЕННАЯ ОБРАБОТКА ДАННЫХ

* * *

ПОСТРОЕНИЕ РАСПРЕДЕЛЕННЫХ МОДЕЛЕЙ В СИСТЕМЕ SIMINTECH

Методические указания
по выполнению лабораторных работ
для бакалавров направления 09.03.02
"Информационные системы и технологии"

Санкт-Петербург
2018

Рассмотрено и рекомендовано к изданию
Институтом леса и природопользования
Санкт-Петербургского государственного лесотехнического университета
16 февраля 2018 г.

Составители:
кандидат технических наук, доцент **С.П. Хабаров**,
кандидат технических наук, доцент **М.Л. Шилкина**

Отв.редактор:
кандидат технических наук, профессор **А.М. Заяц**

Рецензент:
кафедра информационных систем и технологий СПбГЛТУ

Построение распределенных моделей в системе SimInTech: методические указания / сост.: С.П. Хабаров, М.Л. Шилкина − СПб.: СПбГЛТУ, 2018. − 122 с.

Пособие дает представление о компьютерном моделировании автоматических систем, содержит краткое описание среды динамического моделирования технических систем SimInTech, методические указания к лабораторным работам, целевое назначение которых - изучить и освоить методы компьютерного проектирования автоматических систем и построения распределенных моделей таких систем, использующих общесистемную базу данных.

Предназначено для бакалавров направления 09.03.02, обучающихся по специальности «Информационные системы и технологии», а также магистров направления 09.04.02 и студентов других специальностей, связанных с прикладной математикой и информатикой.

Ил. 160. Библиогр. 4 назв.

Темплан 2018 г. Изд. № 18

ВВЕДЕНИЕ

SimInTech — система автоматизированного проектирования логико-динамических систем, описываемых во входо-выходных отношениях, в виде систем обыкновенных дифференциальных уравнений и/или дифференциально-алгебраических уравнений, а также описываемых в виде расчётных («нодализационных») схем для специализированных решателей (расчётных кодов) термогидродинамических и электромеханических процессов.

Название SimInTech является сокращением от перевода на английский язык оригинального названия технологии программного комплекса «МВТУ» (Моделирование в Технических Устройствах, Simulation In Technic). SimInTech – это альтернатива таким программным продуктам как Matlab/Simulink (MathWorks), Dymola (Dassault), AMESim (Siemens), SCADE (ANSYS), VisSim (Visual So-lutions), SimulationX (ESI ITI GmbH), Simscape, Modelica, LabVIEW (National Instruments) и другим.

SimInTech предназначена для детального исследования и анализа нестационарных процессов в ядерных и тепловых энергоустановках, в системах автоматического управления, в следящих приводах и роботах, и, вообще говоря, в любых технических системах, описание динамики которых может быть представлено в виде системы дифференциально-алгебраических уравнений и/или реализовано методами структурного моделирования. Основными направлениями использования SimInTech являются создание моделей, проектирование алгоритмов управления [2], их отладка на модели объекта, генерация исходного кода на языке Си для программируемых контроллеров.

Для SimInTech созданы и разрабатываются модули расширения, позволяющие создавать модели на базе специализированных расчетных кодов и интегрировать их в комплексные модели и проекты [1].

Данный программный комплекс может:

- использоваться для моделирования нестационарных процессов в физике, в электротехнике, в динамике машин и механизмов, в астрономии и т.д., а также для решения нестационарных краевых задач (теплопроводность, гидродинамика и др.);
- функционировать в мультикомпьютерных моделирующих комплексах, в том числе и в системах удаленного доступа к технологическим и информационным ресурсам;
- функционировать как САПР при групповой разработке и сопровождении жизненного цикла изделия (проекта) при модельно-ориентированном подходе к проектированию.

Областями применения системы SimInTech являются:

- Проектирование автоматических регуляторов.
- Проектирование алгоритмов логико-дискретного и функционально-группового управления.
- Проектное расчетное обоснование алгоритмов автоматизированных систем управления технологическими процессами.
- Программно-инструментальное средство разработки и функционирования модели АСУ ТП в составе полномасштабной модели объекта управления.

К основным функциональным возможностям SimInTech относятся:

- Универсальная модульная графическая оболочка, позволяющая реализовать возможность подключения сторонних модулей и создания расчётных схем различного класса, в сочетании с возможностью создания видеокадров для отображения и управления расчётом.
- Возможность моделирования систем различного типа в рамках единой оболочки с обменом данными между различными расчётными программами через единую базу сигналов и синхронизацией.
- Моделирование и анализ систем управления.
- Генерация кода для исполнительной системы.
- Возможность удалённой отладки программ, собранных на целевой системе по сгенерированному коду, с выводом данных непосредственно на исходную схему алгоритма.

Программный комплекс SimInTech реализует следующие режимы работы:

- *Моделирование* — режим работы, который обеспечивает:
- моделирование нестационарных процессов в непрерывных, дискретных и гибридных технических системах, в том числе и при наличии обмена данными (синхронный или асинхронный) с внешними программами и устройствами;
- редактирование параметров структурной схемы и расчета в режиме "on-line";
- расчет в реальном времени или в масштабированном модельном времени;
- рестарт, архивацию и воспроизведение результатов моделирования.
- *Оптимизация* — позволяет решать задачи:
- параметрической оптимизации САУ и идентификации опытных данных;

- синтеза оптимальных регуляторов и оптимального управления в многокритериальной постановке при наличии ограничений на значения динамических переменных, управляющих воздействий, параметров элементов системы автоматического управления, функционалов качества.
- *Анализ* — режим работы, который обеспечивает:
- расчет амплитудно-фазовых частотных характеристик для любой линейной и большинства нелинейных систем (ЛАХ, ФЧХ, различные годографы и др.);
- расчет коэффициентов, полюсов и нулей передаточных функций.
- *Контроль и управление* — этот режим позволяет:
- создавать электронные (виртуальные) аналоги измерительных приборов и управляющих устройств — для оперативного контроля и управления переходными процессами в объекте моделирования;
- выполнять статистическую обработку сигналов (в том числе и внешних), основанную на быстром преобразовании Фурье.
- *Генерация кода* — данный режим очень важен, так как обеспечивает:
- автоматическое создание кода на языке Си или на языке промышленных контроллеров ST по набранным в SimInTech схемам алгоритмов;
- сборку расчетного модуля, загрузку его на целевую систему, отладку выполнения модуля (алгоритма) на внешней целевой системе;
- автоматизированную организацию межприборного обмена в случае программирования нескольких приборов одновременно.

SimInTech не имеет аналогов среди отечественного ПО. За рубежом его аналогами являются такие программным продукты, как SimuLink, MATRIX, VisSim, SimulationX и некоторые другие.

Версии SimInTech 1.1 соответствует версия ПК «МВТУ» 4.0. Более новая версия SimInTech 1.2, особенно ее графическая часть, использует технологию Direct2D, минимально необходимой операционной системой для которой является Windows 7. Разработчиком и распространителем системы является ООО «ЗВ Сервис», на сайте которой http://simintech.ru/ представлены подробные данные о системе и большой объем справочной информации.

Авторы благодарят ООО "САРК-ИС" и его генерального директора К.С.Хабарова за финансовую поддержку издания данного пособия.

1. ОСНОВЫ РАБОТЫ В СРЕДЕ SIMINTECH

Лабораторная работа № 1

Целью данной лабораторной работы является:

1. Получить общее представление о системе SimInTech.
2. Познакомиться с основными методами и приемами работы в этой среде.
3. Изучить возможности встроенной в пакет справочной системы.
4. Рассмотреть ряд примеров проектов, включенных в состав пакета.

Среда динамического моделирования технических систем SimInTech является гибким и мощным средством для разработки сложных расчетных моделей. Создание таких моделей требует специального инструмента для подготовки модели и возможности ее разделения на более простые части для удобства коллективной работы. В комплексной модели могут быть объединены расчеты физических процессов, рассчитываемых по различным расчетным кодам (тепло-гидравлическим, электрическим и др.).

1.1. Концепция создания комплексной модели объекта в среде SimInTech

Одной из основных особенностей использования SimInTech для создания 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 рис. 1.1.

При создании модели с использованием базы данных сигналов возможно разделение сложной комплексной модели на произвольное ко-

Свойство:

- Количество портов – количество входных сигналов блока.

Примечания:

- Блок формирует *один* выходной сигнал.
- Именованные свойства могут быть заданы как *локальные* переменные модели (субмодели) во вкладке «Параметры» и как *глобальные* сигналы проекта при помощи пункта главного меню «Сервис → Сигналы», или как *внешние* сигналы проекта при помощи подключаемой базы сигналов.

Блок «Демультиплексор»

Этот блок векторизован, реализует “расщепление” векторного входного сигнала в отдельные выходные сигналы. Выходами могут быть как скалярные, так и векторные (“многожильные”) сигналы.

По умолчанию данный блок реализует демультиплексирование 2-х “жильного” входного векторного сигнала в скалярные сигналы:

$$y_1(t) = u_{1,1}(t),$$

$$y_2(t) = u_{1,2}(t),$$

где $y_1(t)$, $y_2(t)$ – первый и второй выходной *скалярный* сигнал блока, а $u_{1,1}(t)$, $u_{1,2}(t)$ – 1-й и 2-й элементы входного *векторного* сигнала.

Свойства:

- Массив размерностей выходов – массив целых чисел, указывающих размерности векторов каждого из выходных сигналов блока.

Примечания:

- Блок обрабатывает *один* входной сигнал.

Именованные свойства могут быть заданы как *локальные* переменные модели (субмодели) во вкладке «Параметры», как *глобальные* сигналы проекта при помощи пункта главного меню «Сервис → Сигналы», или как *внешние* сигналы проекта при помощи подключаемой базы сигналов.

6. ОРГАНИЗАЦИЯ РАСПРЕДЕЛЕННЫХ МОДЕЛЕЙ С ОБЩЕСИСТЕМНОЙ БАЗОЙ ДАННЫХ

Лабораторная работа № 6

Целью данной лабораторной работы является:

1. Знакомство с взаимодействием проектов через общую базу данных сигналов.

2. Построение распределенной модели с обменом данными через общую базу.
3. Изучение методов сетевого расчета в ручном и автоматизированном режимах.
4. Получение навыков работы с векторной обработкой сигналов.

Одной из основных особенностей использования SimInTech для комплексных моделей является идеология использования «Базы данных сигналов» - структурированную файловую базу данных, содержащую в себе константы и переменные (сигналы), обеспечивающих обмен расчетными значениями между расчетными схемами в единой модели. База данных SimInTech является объектной и позволяет:

- объединить нескольких расчетных схем в единую модель;
- обеспечить векторную обработку сигналов для типовых алгоритмов управления;
- выполнить объектно-ориентированное проектирование моделей систем;
- автоматизировать создание и обработку переменных в комплексных моделях.

Таким образом, в среде SimInTech, кроме сигналов проекта и глобальных переменных языка программирования, можно использовать внешнюю по отношению к файлу проекта базу данных (БД). Она подключается как динамическая библиотека. Также база данных может содержать информационные поля (строковые константы), которые не используются для целей моделирования. Элементарной единицей базы сигналов является сигнал. Сигнал – переменная одного из доступных типов данных, несущая какую-либо информацию. В стандартную поставку SimInTech включена БД сигналов SDB, подключение которой к файлу проекта выполняется на вкладке «Настройки» в параметрах расчета проекта, а для работы с БД используется редактор, общий вид которого представлен на рис. 6.1.

База данных сигналов SDB представляет собой структурированную базу данных, содержащую три уровня, а именно «Категории», «Группы сигналов» и «Сигналы групп».

- Категория − является прообразом, шаблоном или, в терминах программирования, классом для объектов базы данных.
- Группа сигналов − является логической единицей базы данных и всегда представляет собой объект какой-либо категории.

- Сигнал – является физически существующей в области памяти единицей: одной из переменных группы, именованных и типизированных.

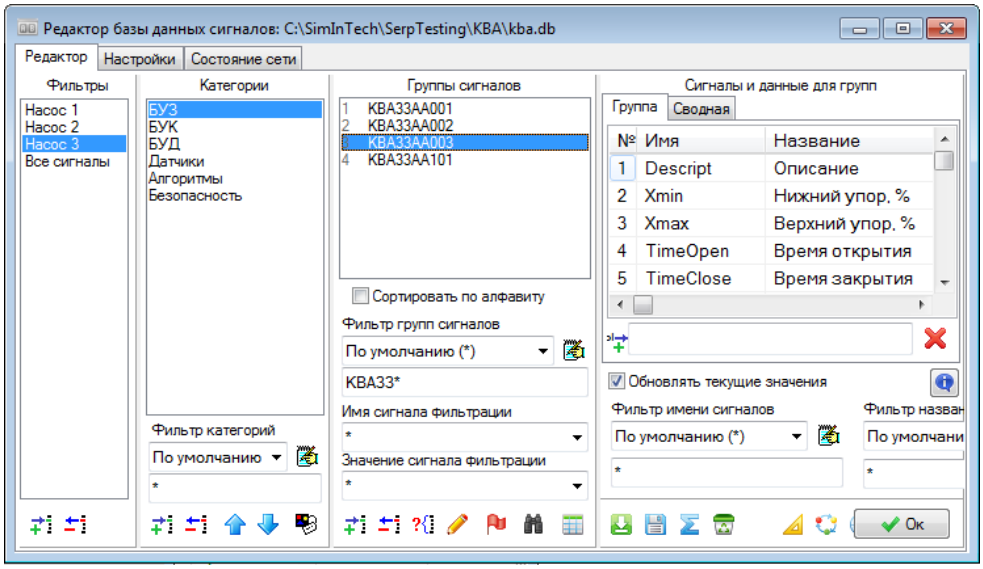

Рис. 6.1.Общий вид редактора базы данных сигналов.

Таким образом, категории создают архитектуру БД, группы сигналов – наполнение, а сигналы составляют список сигналов базы данных, который формируется и может быть использован в проекте.

В этом разделе основное внимание будет уделено использованию общесистемной базы данных для взаимодействия отдельных проектов в распределенных моделях, работающих как в рамках пакета проектов на одном компьютере, так и при их совместной работе на разных узлах сети. Поэтому тестовый пример будет максимально простым и посвящен реализации модели ручного управления клапанами для залива и осушения цистерн.

6.1. Формирование модели ручного управления клапанами

Ранее были рассмотрены достаточно сложные модели по управлению уровнем жидкости в цистерне. Здесь остановимся только на простейшей, но распределенной модели ручного управления входными и выходными клапанами цистерн, которая должна использовать базу данных SimInTech. Но прежде чем формировать общесистемную базу данных сигналов надо определиться со структурой модели.

На первом этапе построим комплексную модель, реализуемую на одном компьютере, которая представляет собой пакет проектов. Пакет включает в себя три проекта: модель панели управления, модель форми-

рования управляющих воздействий на клапаны и модель объекта управления. С этой целью:

- создадим новую папку tank_db_1, а в ней пакет проектов tank_db_pak.pak из трех новых проектов − tank_db_main.prt, tank_db_control.prt и tank_db_object.prt.
- Особо отметим, что во всех проектах надо использовать блоки «Запись в список сигналов» и «Чтение из списка сигналов» для тех сигналов, которыми проекты обмениваются между собой. На первом этапе это будут пустые блоки, в которых отсутствует определение каких-либо свойств.
- Начнем с модели панели управления. Схема проекта содержит всего две кнопки, которые связаны с блоками записи сигналов в список (рис. 6.2). Приведенную на рисунке шкалу на этом этапе можно не реализовывать. Здесь она использована как напоминание о том, что на панель управления должны приходить с сигналы с объекта управления.

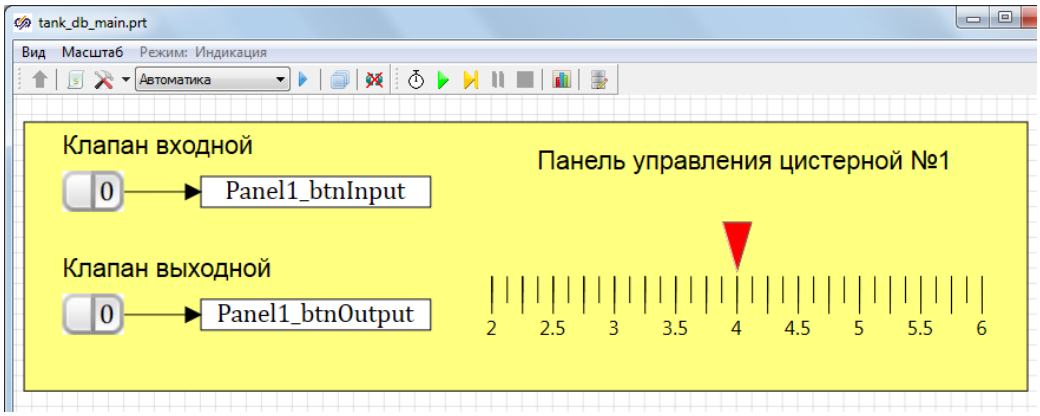

Рис. 6.2.Схема проекта tank_db_main.prt.

- Сигналы от кнопок панели управления поступают в алгоритм формирования управляющих воздействий на клапаны, который реализует проект tank_db_control. (рис. 6.3). В данном примере это простейшая блокировка одновременной работы двух клапанов.
- Модель объекта управления может быть получена путем копирования в новый проект набора блоков из любого ранее используемого проекта (рис. 6.4).

Данная комплексная модель может успешно работать только в том случае, если обеспечено взаимодействие между собой всех проектов. Одним из подходов к реализации этой задачи в SimInTech является возможность подключать к любой схеме базу данных сигналов. Она служит для

структурного хранения глобальных констант и переменных, а также для организации взаимодействия между расчетными схемами.

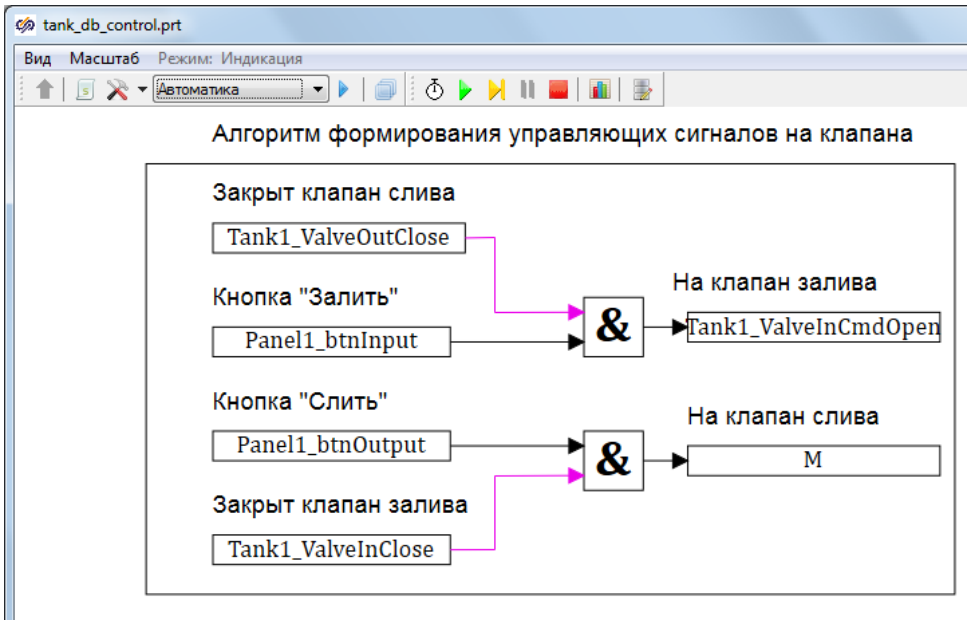

Рис. 6.3.Схема проекта tank_db_control.prt.

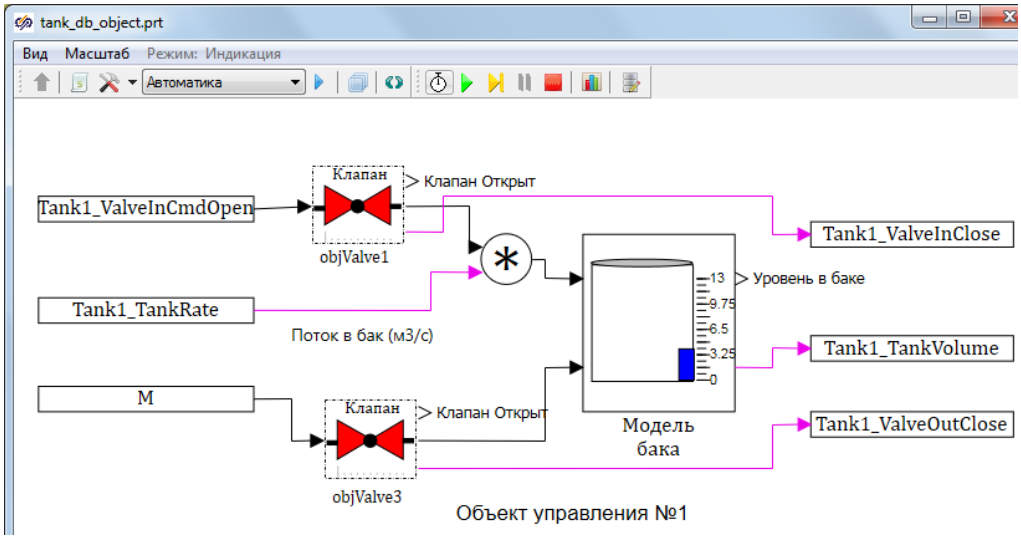

Рис. 6.4.Схема проекта tank_db_object.prt.

Для совместной работы нескольких проектов, находящихся в разных файлах, надо, чтобы они использовали одну и ту же базу сигналов, структура которой учитывала бы особенность каждого из проектов.

6.2. Подключение базы данных к пакету проектов

Будем считать, что базовым у нас является проект tank_db_main.prt, находясь в котором пользователь имеет полные права на создание и модификацию структуры общесистемной базы данных. Но прежде всего на-

до обеспечить подключение базы данных к этому проекту и выполнить следующую последовательность действий.

- Проверить, что установлен режим разработчика, а затем, открыв окно параметров расчета проекта, выбрать вкладку «Настройки» (рис. 6.5).

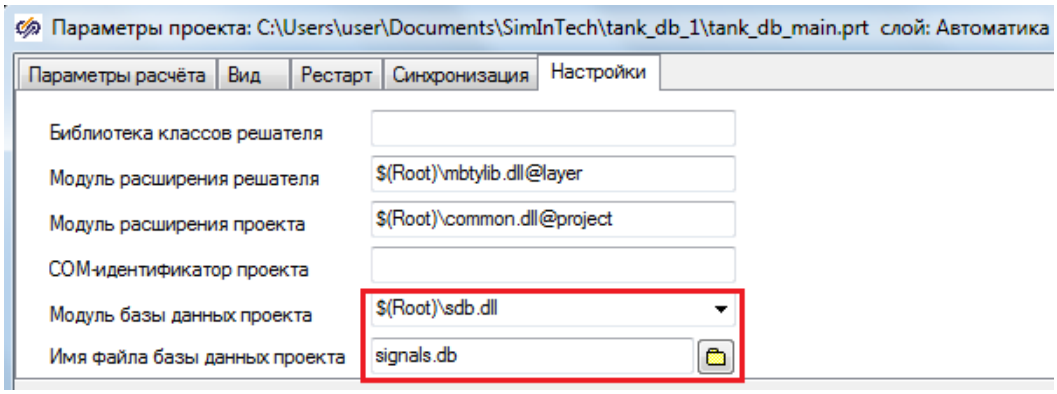

Рис. 6.5. Вкладка «Настройки» окна «Параметров проекта».

- В поле «Модуль базы данных проекта» ввести: $(Root)\sdb.dll, где sdb.dll – имя динамической библиотеки программного модуля базы данных сигналов, поставляемой в составе стандартного дистрибутива;
  а значение $(Root) – служебное имя, указывающее на папку C:\SimInTech\bin.
  В поле «Модуль базы данных проекта» можно ввести C:\SimInTech\bin\sdb.dll, что равнозначно $(Root)\sdb.dll.
- В поле ввода «Имя базы данных проекта» ввести произвольное имя файла для сохранения базы данных. В рамках рассматриваемого примера присвоим имя базе сигналов signals.db, однако оно может быть любым. Если не указан полный путь к файлу, то он располагается в том же каталоге, что и файл проекта.
- Закрыть окно нажатием «Ок» и выполнить сохранение текущего проекта, нажав кнопку «Сохранить проект» в главном меню SimInTech.

После сохранения проекта с подключенной к нему базой, файл signals.db появится на диске в каталоге проекта, что можно проверить просмотром этого каталога. Кроме того, для проекта с подключенной базой становится в главном меню доступна опция «Инструменты → База данных...». Выбрав ее, откроется окно редактора базы данных сигналов (рис. 6.6).

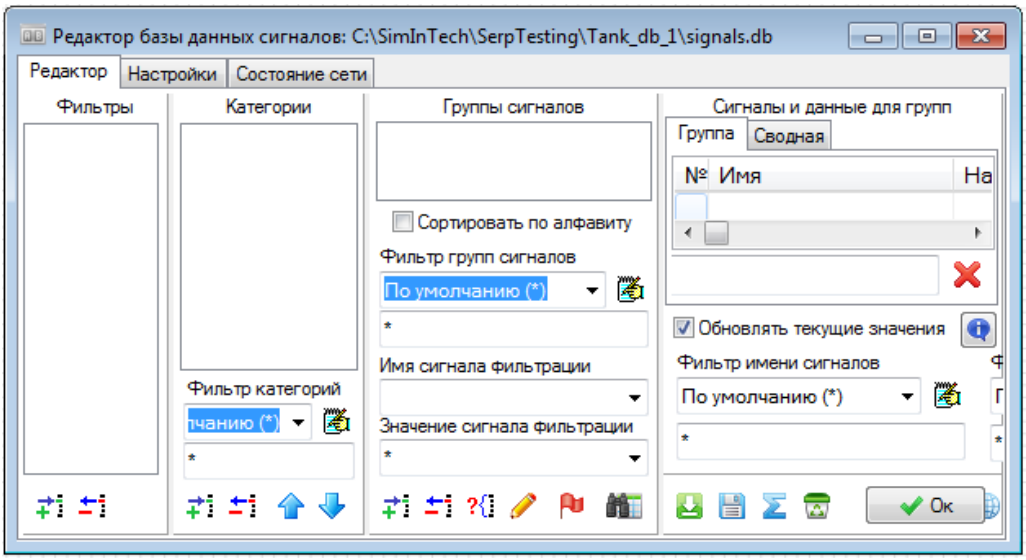

Рис. 6.6. Доступ к редактору БД из проекта tank_db_main.prt.

Это окно пока пустое, его заполнять будем позднее. Но прежде чем закрывать редактор, следует убедиться, что настройки предполагают автоматическое сохранение базы данных при работе с этим проектом. Для этого надо перейти на вкладку «Настройки» и установить галочки в пунктах «Сохранять базу» и «Резервировать БД» (рис. 6.7).

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 с расширениями .1 .2 .3 и так далее.

- Установив нужные опции, надо вернуться на закладку «Редактор» и закрыть окно «Редактор базы данных» нажатием кнопки «Ок».
- После этого надо сохранить проект. Данные установки позволяют автоматически сохранять базу данных проекта при закрытии файла самого проекта.

Отметим, что пользователь в любой момент времени может принудительно сохранить базу данных, не закрывая проект. Для этого достаточно 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.

- Замечание 1. При новом подключении проекта к уже существующей базе, выполняя сохранение этого проекта надо отве-

тить «Нет» на запрос об изменении базы. Иначе существующая база будет перезаписана пустым файлом.

- Замечание 2. База сигналов считывается из файла только при загрузке проекта, а не при прописывании файла базы сигналов в проекте.

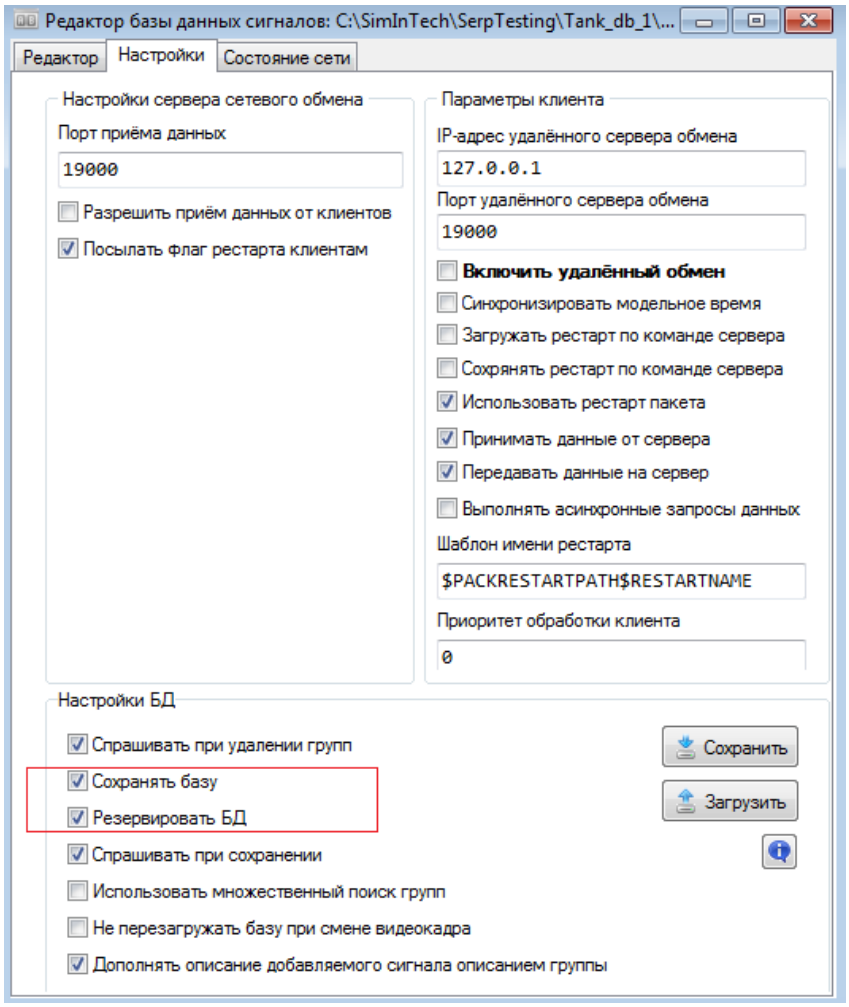

Рис. 6.7. Окно настройки БД, подключенной к проекту tank_db_main.

После привязки проекта tank_db_main к базе signals.db, аналогичные действия надо выполнить и во всех других проектах пакета. Однако, так как базу использует ряд проектов, то нужна еще дополнительная настройка возможности ее сохранения в разных проектах.

Это требуется для того, чтобы изменения в базу сигналов вносились только из какого-то одного проекта, чтобы избежать конфликтов записи, некорректной перезаписи файла или потери информации, сохраненной в него из другого проекта. Поэтому, после подключения базы к проектам tank_db_object.prt и tank_db_control.prt, рекомендуется в этих проектах опции «Сохранять базу» и «Резервировать БД» отключить. То есть, все из-

менения в общесистемной базе данных будут выполняться только в основном проекте.

6.3. Заполнение базы данных и связь блоков модели с сигналами базы данных

Что касается структуры самой базы данных, то обычно ее объектно-ориентированный характер описывает все элементы и узлы модели исследуемой системы с полным набором их параметров, характеристик и сигналов. Основное назначение рассматриваемого примера состоит в том, чтобы показать возможность взаимодействия разных проектов между собой через единую базу данных. Поэтому далее предлагается использовать упрощенную структуру БД, описывающую только те объекты, которые соответствуют основным структурным блокам проектов, взаимодействующих между собой в рамках распределенной модели.

Если обратить внимание на исследуемую модель и проанализировать все выделенные в ней блоки чтения и записи в память, то можно установить, что для описания всех взаимодействий между проектами достаточно всего двух категорий объектов − «Панели» (ObjectPanel) и «Объекты Управления» (ObjectTank). Третий проект не имеет собственных сигналов, а использует сигналы двух первых объектов.

Чтобы сформировать БД для исследуемой модели, надо открыть проект tank_db_main и активировать редактор базы данных. Подробную инструкцию по работе с ним можно найти в справочной системе SimInTech, в частности, в разделе 7 − ”Общие сведения о базе данных сигналов SDB”. Для редактирования списков категорий, групп сигналов, фильтров, а также сигналов и данных групп, под каждой панелью предусмотрены кнопки «Добавить» и «Удалить». Используя кнопку «Добавить», создадим две новые категории (рис. 6.8).

В процессе добавления и нажатия кнопки «Добавить категорию» в списке категорий появляется запись «Новая категория». Двойное нажатие на элементе «Новая категория» откроет окно «Шаблон категории», где надо задать имя категории и заполнить таблицу шаблонных сигналов. Расположенные в нижней части окна кнопки редактирования таблицы сигналов позволяют: добавить, удалить сигнал, очистить список сигналов, загрузить список сигналов из файла или сохранить в файл, копировать сигнал в буфер памяти, вставить сигнал из буфера памяти, переместить сигнал в таблице выше/ниже. При добавлении в проектируемую базу данных каждой из новых категорий окна «Шаблон категории» будут иметь вид, аналогичный тому, что представлен на рис. 6.9.

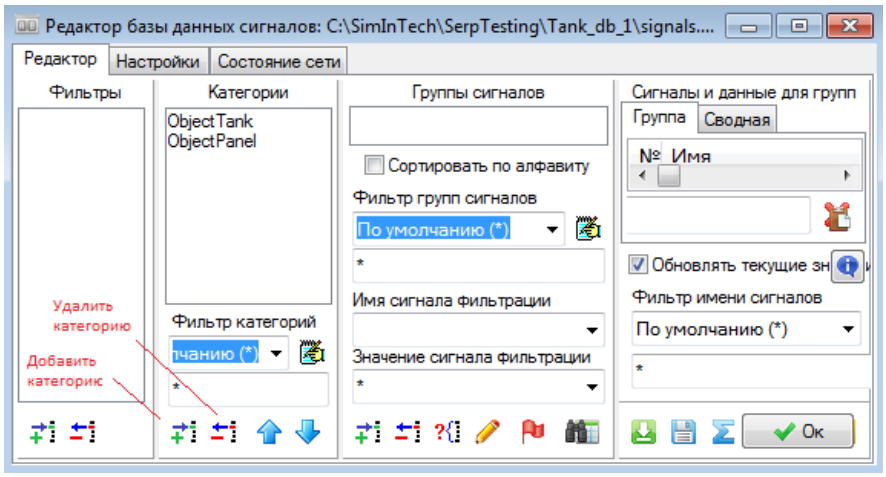

Рис. 6.8. Создание двух новых категорий в БД.

Выполненные действия аналогичны созданию двух типов объектов, а также описанию всех свойств каждого из этих объектов. При этом каждое из свойств может описываться разным типом данных и иметь различные способы обработки (расчета). Описав два типа объекта, можно перейти к созданию конкретных экземпляров этих объектов.

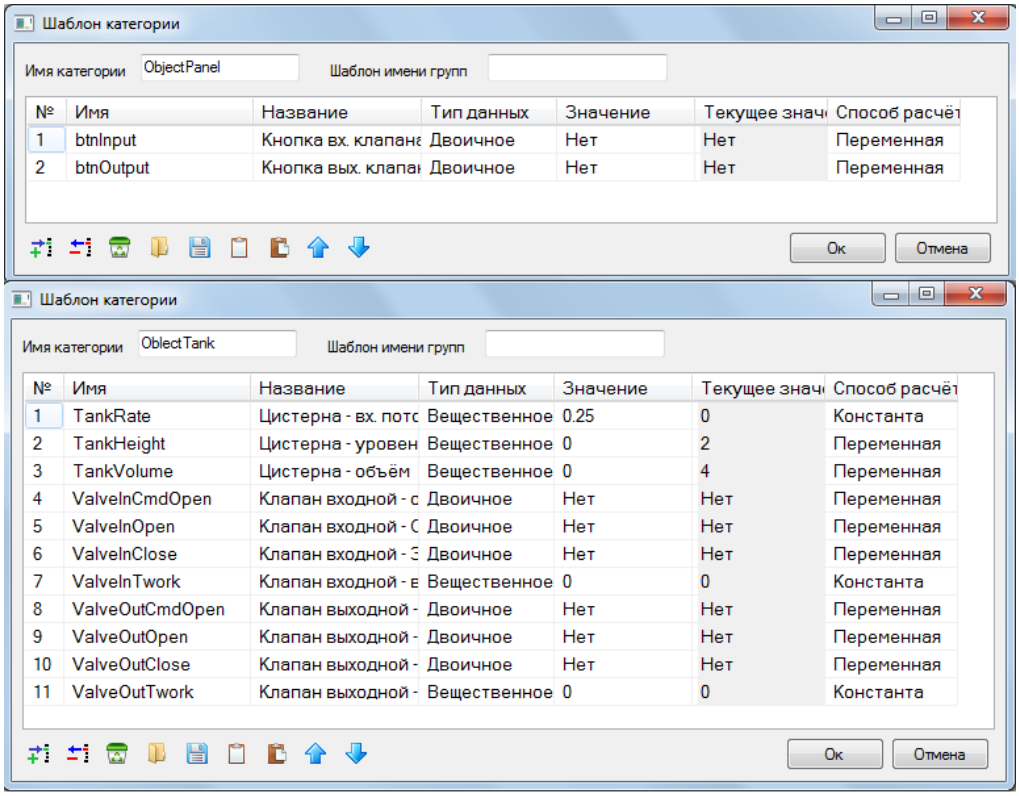

Рис. 6.9. Описание категорий ObjectPanel и ObjectTank.

Для каждой категории пользователь может добавить в базу данных неограниченное количество групп сигналов. Например, создав категорию панелей, можно добавить в базу данных несколько панелей данного типа, хотя пока в расчетной модели используется только одна панель. Для этого необходимо:

- выделить название категории в панели категорий, например, ObjectPanel. • Нажать кнопку «Добавить таблицу» в панели групп сигналов;
- откроется окно «Создание новых групп», в котором надо ввести имена групп сигналов данной категории, которые необходимо внести в базу данных (рис. 6.10);
- закрыть диалоговое окно «Создание новых групп» нажатием кнопки «Ок».

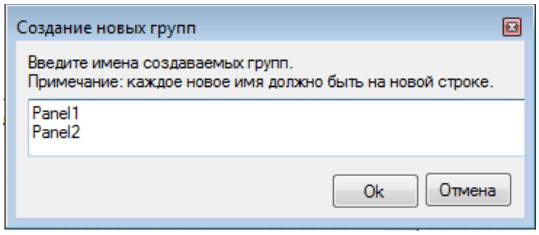

Рис. 6.10. Создание групп сигналов категории ObjectPanel.

Аналогично надо создать две группы сигналов, например Tank1 и Tank2, для категории ObjectTank. После заполнения БД категориями и группами станет доступным переключение между ними, путем выбора нужной категории и группы сигналов. При этом все сигналы выбранной группы будут отображаться в правой части редактора (рис. 6.11).

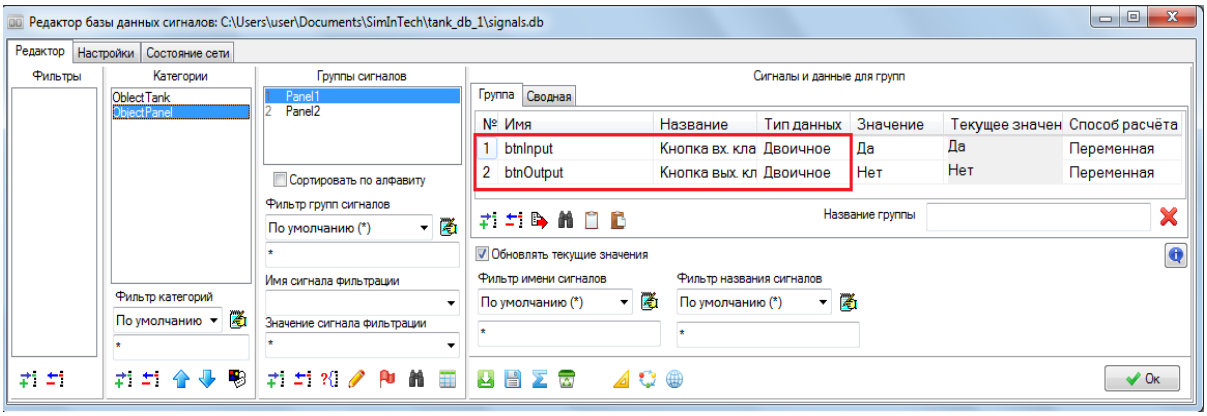

Рис. 6.11. Состав сигналов группы Panel1 категории ObjectPanel.

После окончания заполнения базы данных сигналов, надо сохранить ее содержимое и приступить к связи блоков пакета проектов с сигналами базы данных. В качестве примера рассмотрим проект tank_db_main, где имеется два выходных сигнала от кнопок, которые подключены к блокам «Запись в список сигналов». Чтобы они могли взаимодействовать с блоками других проектов, надо некоторые их свой-

ства подключить к БД. Так, для кнопки управления входным клапаном надо выполнить набор следующих операций (рис. 6.12).

1. На схеме проекта надо выделить нужный блок «Запись в список сигналов» и открыть окно его свойств.

2. В окне свойств элемента выбрать свойство signals («Имена сигналов»).

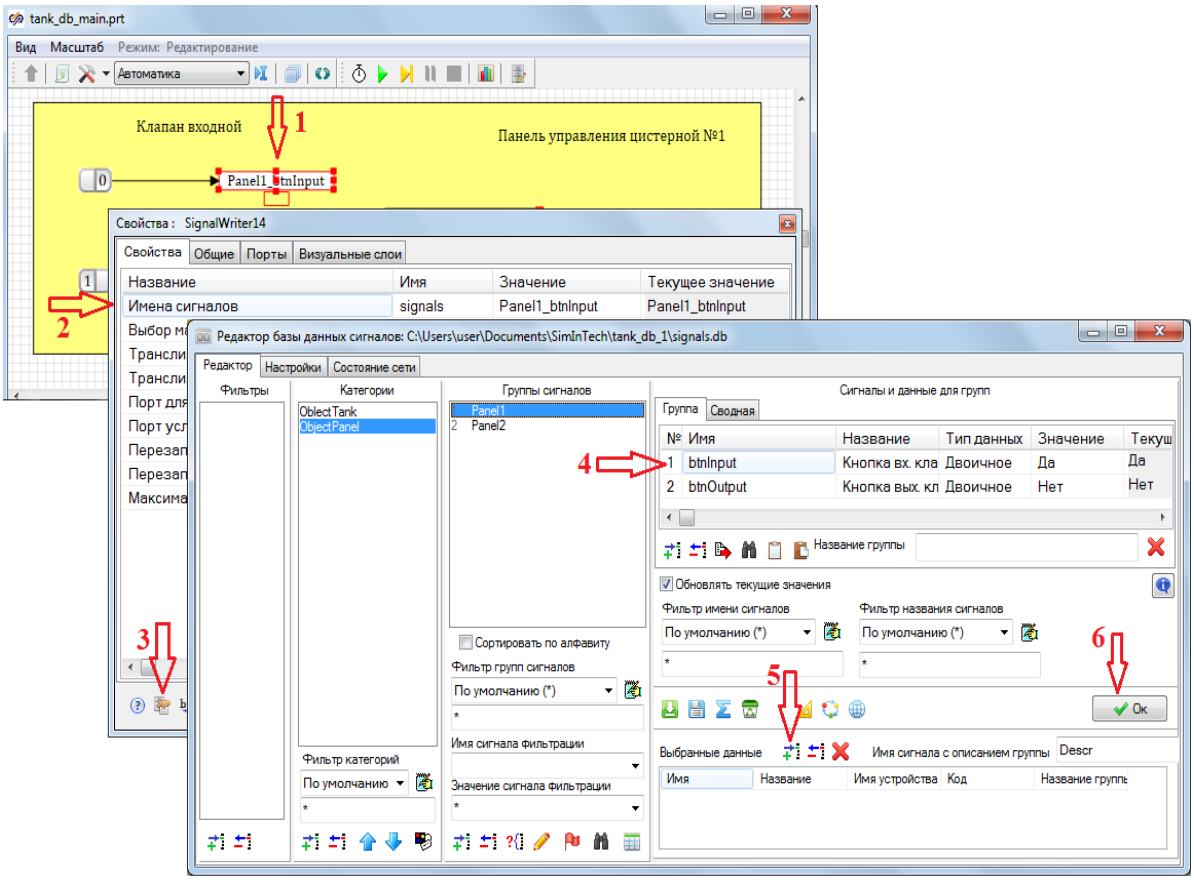

Рис. 6.12. Подключение блока «Запись в список сигналов» к сигналу БД.

3. Нажать в нижней части окна кнопку «Выбрать сигнал из базы», при этом откроется окно редактора базы данных.

4. В этом окне надо выбрать соответствующий сигнал соответствующей группы из соответствующей категории. В нашем примере это сигнал btnInput из группы Panel1 категории ObjectPanel.

5. После этого необходимо добавить его в панель «Выбранные данные», нажав в этой панели кнопка «Добавить».

6. Завершить операцию нажатием кнопки «Ок» в этом, и всех открытых окнах.

После этого вид блока «Запись в список сигналов» в схеме проекта будет иметь вид, аналогичный тому, что приведен на рис. 6.13. Обра-

тите внимание, что сигнал базы данных, который подключен к блоку схемы, SimInTech назвал как <Имя группы>_<Имя сигнала>.

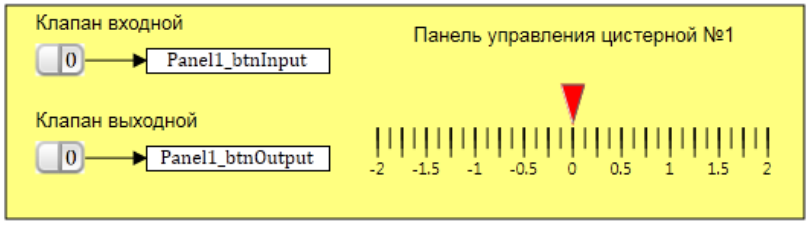

Рис. 6.13. Схемы проекта после подключения к сигналам БД.

Принимая во внимание такое именование в SimInTech сигналов базы данных, можно для второго блока «Запись в список сигналов» открыть окно его свойств, и в свойстве signals сразу или с помощью редактора ввести текст Panel1_btnOutput.

Закончив с проектом tank_db_main, можно перейти к tank_db_object и tank_db_control, и выполнить подключение их блоков записи и чтения в список сигналов к сигналам базы данных (рис. 6.14).

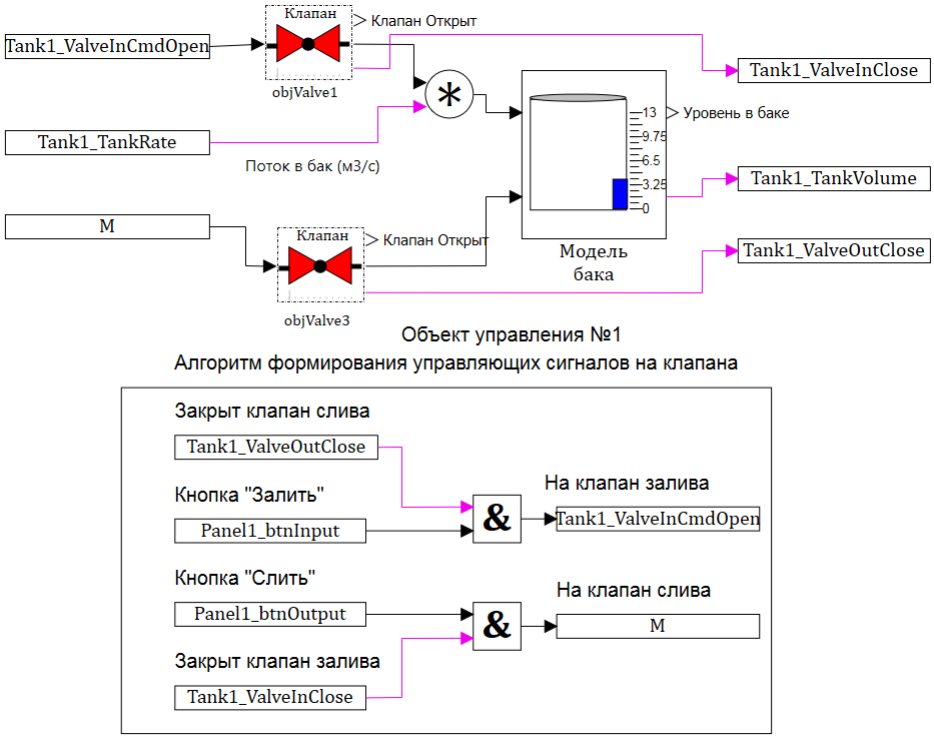

Рис. 6.14. Схемы проектов после подключения к сигналам БД.

Обратите внимание, что сигнал Tank1_ValveOutCmdOpen на схеме не отобразился и блок не изменил свое обозначение. Это связано с тем, что SimInTech отображает в блоках только те сигналы, символиче-

ские имена которых содержат не более 20 символов, а имя указанного выше сигнала состоит из 21 символа.

На текущий момент все необходимые преобразования комплексной модели по связи блоков с сигналами базы данных выполнены, и модель может быть запущена в работу. Для этого достаточно загрузить проект tank_db_pak.pak и нажать кнопку запуска проекта. Если все сделано было верно, то пакет должен работать, обмениваясь между проектами сигналами общесистемной базы данных. Причем об изменении всех сигналов в процессе работы пакета можно следить, если в любом из проектов дополнительно открыть окно редактора базы данных сигналов (рис. 6.15).

Сигналы БД могут использоваться в задании свойств не только типовых блоков, но и в любых графических компонентах, в том числе и в примитиве типа «Bar» (рис. 6.16). На рисунке видно, что нижний и верхний пределы шкалы, а также отображаемое значение через значение сигнала БД Tank1_TankVolume.

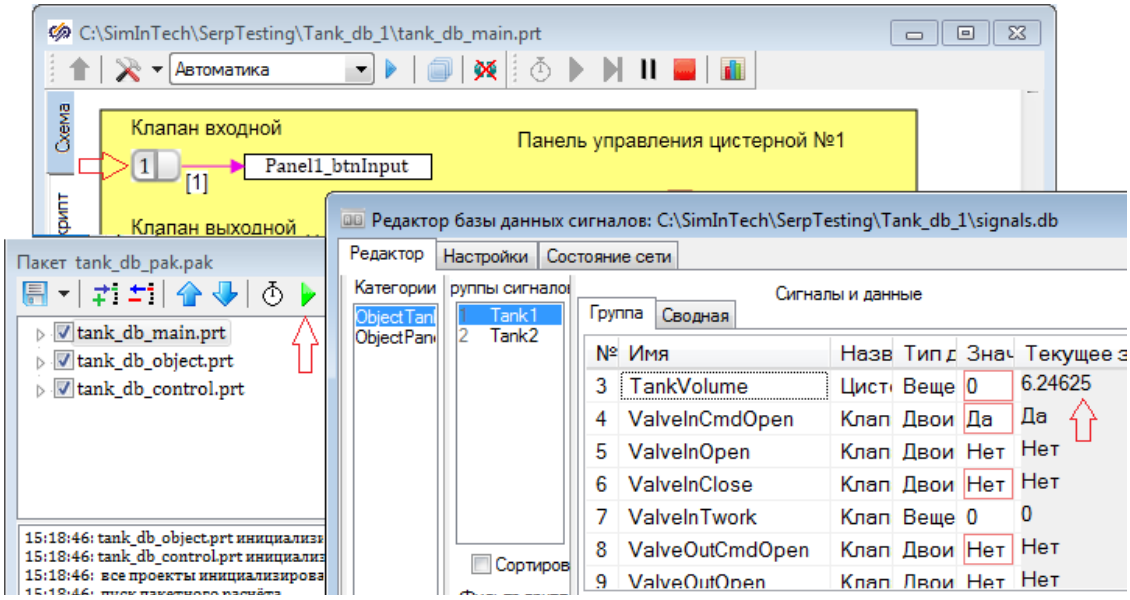

Рис. 6.15. Просмотр текущего состояния сигналов в редакторе БД.

Данный примитив на схеме основного проекта – это подвижная шкала, отображающая текущий объем цистерны, вычисленный в проекте tank_db_object и переданный сюда через общесистемную базу данных. Одновременно с этим сигнал с кнопки этого проекта через БД передается в проект tank_db_control, а оттуда в проект tank_db_object.

Круг обмена данными между проектами замкнулся.

Свойства : Bar

Общие | Визуальные слои

Название	Имя	Значение
Нижний предел	MinValue	Tank1_TankVolume - 2
Верхний предел	MaxValue	Tank1_TankVolume + 2
Шаг	Step	0.5
Число делений на шаг	DivCount	2
Высота штриха (% от размеров)	NumHeight	70
Высота деления (% от размеров)	DivHeight	30
Показывать деления	ShowDivs	Да
Показывать числа	ShowNums	Да
Шрифт	Font	Arial Narrow
Число больших делений на шаг	LargeDivCount	2
Толщина малого штриха	SmallWidth	1
Толщина большого штриха	LargeWidth	1
Штрихи наоборот	InvTicks	Нет
Высота среднего штриха (% от разм	MidHeight	50
Количество средних штрихов на шаг	MidDivCount	4
Коэффициент прозрачности	Opacity	1
Тип столбца	BarType	Нет
Цвет столбца	BarColor	16711680
Ширина столбца (% от высоты)	BarWidth	40
Тип ручки	ControlType	Треугольник
Трёхмерность ручки	Handle3d	Да
Цвет ручки	ControlColor	255
Толщина ручки	ControlWidth	15
Показывать шкалу	AxisVisible	Да
Режим управления	DirectMode	Нет
Дискретность управления	Discrete	0
Отображаемое значение	Value	Tank1_TankVolume

Рис. 6.16. Основные свойства примитива ”Линейный прибор”.

6.4. Реализация распределенной модели с общей базой данных

В предыдущих разделах описано создание отлаженной комплексной модели в виде пакета проектов, а сейчас преобразуем ее в распределенную сетевую модель, в которой каждый проект мог бы запускаться на своей рабочей станции. Обмен данными между проектами осуществляется через единую базу данных.

Для решения этой задачи сначала определим сетевую конфигурацию той среды, где будет реализована поставленная задача. Ее основу составляют следующие компоненты.

- Компьютер на базе Windows 10 с установленным на нем Oracle VM Virtual Box.
- Две виртуальные машины на базе Windows 7 с системой SimInTech. Они имеют сетевые имена SimInTech-1 (192.168.56.101) и SimInTech-2 (192.168.56.102). Эти имена могут не совпадать с именами машин в менеджере Oracle Virtual Box.

- Еще одна виртуальная машина на базе Windows XP с системой SimInTech. Она имеет сетевое имя SimInTech-3 (192.168.56.103).
- Внутри базового компьютера организована рабочая группа Workgroup, в которую объединены все виртуальные машины.
- Кроме того, на диске D: базового компьютера выделена папка, общедоступная со всех виртуальных машин через vboxsvr.

Проверить настройку сетевой среды можно, если в любой из виртуальных машин открыть проводник Windows и выбрать компонент Сеть (рис. 6.17).

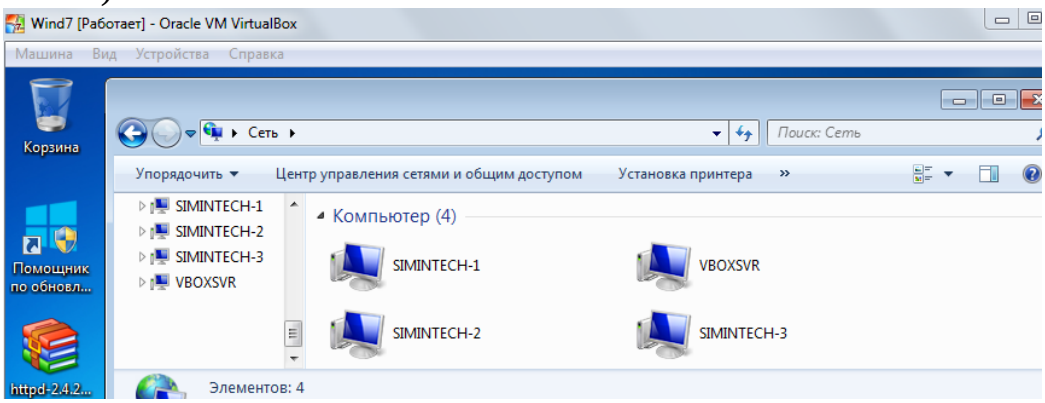

Рис. 6.17. Компьютеры рабочей группы.

При построении распределенной модели планируется, что каждый проект будет запущен на своем компьютере. При этом в настройках каждого проекта будут использованы IP-адреса этих компьютеров. Поэтому желательно уже на начальной стадии убедиться во взаимной доступности всех виртуальных машин по их IP-адресам (рис. 6.18).

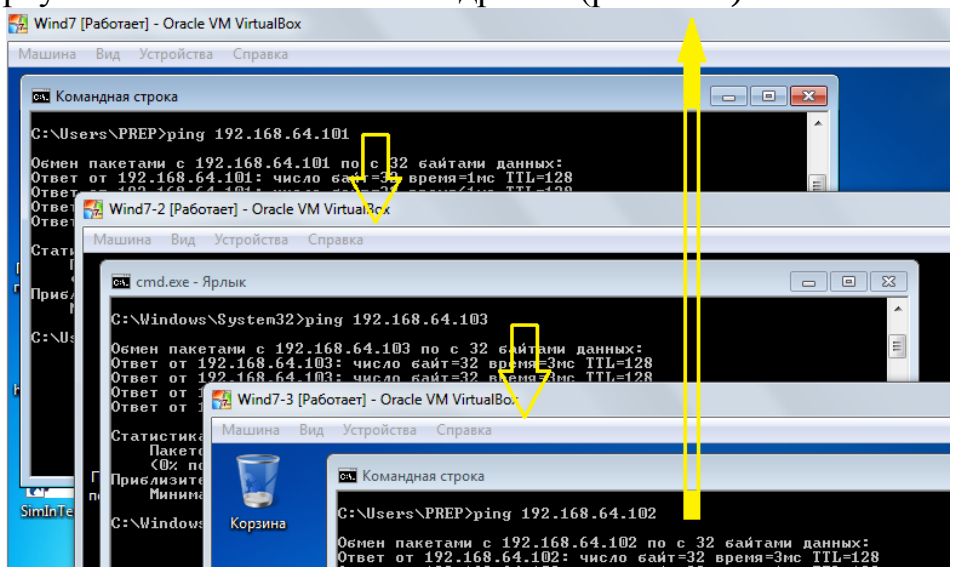

Рис. 6.18. Проверка взаимной доступности виртуальных машин.

Для удобства дальнейшей работы имеет смысл настроить сетевой ресурс с разрешённым доступом к нему со всех трёх виртуальных машин, чтобы можно было настраивать всё на одном компьютере. Для этого используем папку основного компьютера, настроенную в качестве общей во всех виртуальных машинах. Дадим ей имя server и скопируем в нее все файлы работающих проектов, кроме файла пакета. Убедиться в сетевой доступности этого ресурса можно, используя в любой из виртуальных машин проводник Windows (рис. 6.19).

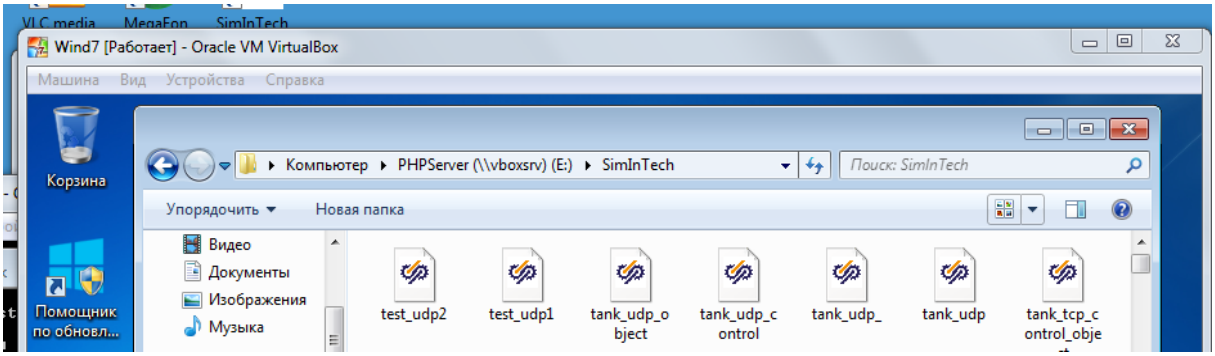

Рис. 6.19. Проверка доступа к общесистемному ресурсу.

Аналогично можно выяснить, что сетевое хранилище данных виртуальной машиной, например SimInTech-2, воспринимается как сетевой диск E: (рис. 6.20).

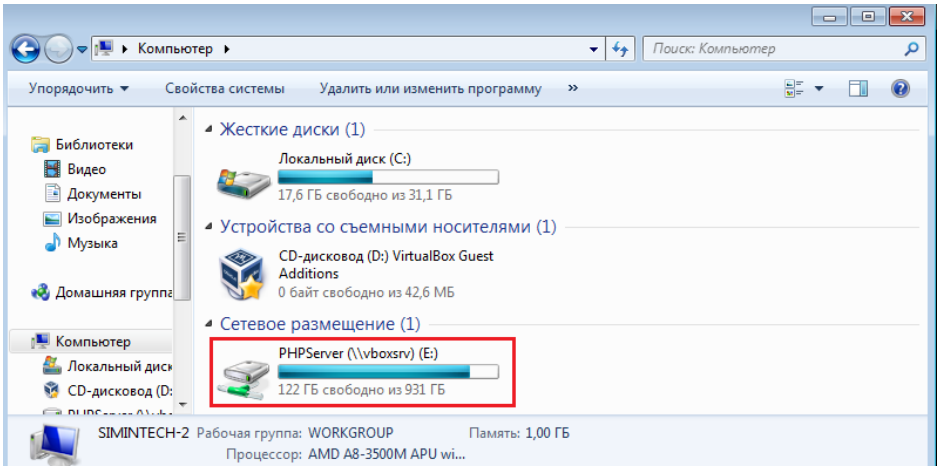

Рис. 6.20. Состав устройств сетевого узла SimInTech-2.

Простейшей сетевой организацией распределенной модели является вариант, когда все файлы хранятся в одном месте, а в процесс работы на каждом из узлов запускается свой проект, загружаемый с общесе-

тевого хранилища данных. При этом один из проектов должен быть настроен как сервер сетевого обмена, а другие как его клиенты (рис. 6.21)

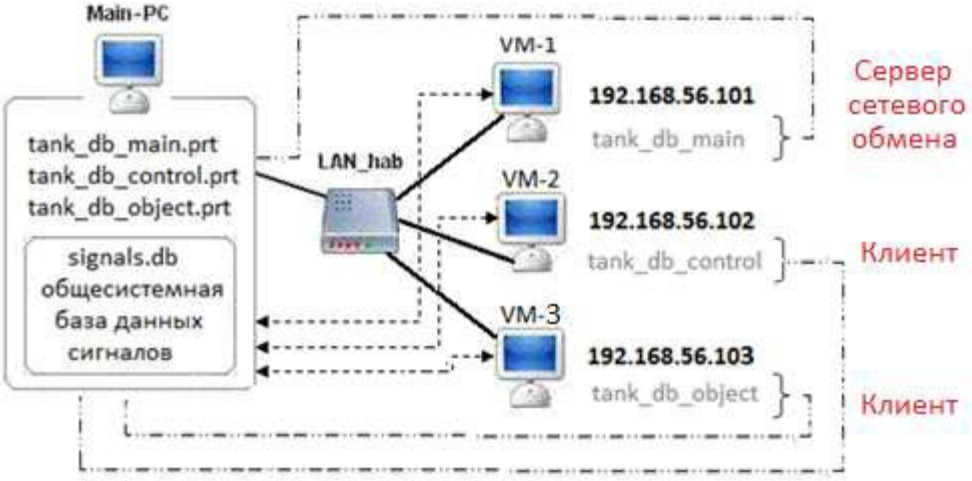

Рис. 6.21. Клиент-серверная структура с общесетевым хранилищем данных.

Закончив настройку и тестирование сетевой среды, следует заняться настройкой распределенной модели для возможности ее работы в сети.

- Для настройки сервера сетевого обмена надо открыть проект tank_db_main.prt, зайти в настройки базы данных и включить опцию «Разрешить приём данных от клиентов». Порт приёма можно не менять (19000), а опция «Включить удалённый обмен» остается выключенной – это настройка для клиентов (рис. 6.22). После этого надо сохранить проект tank_db_main.prt, так как настройки сервера сетевого обмена и параметры клиента хранятся в проекте, а не в файле базы данных.
- Для настройки клиентов надо в проектах tank_db_control.prt и tank_db_object.prt в настройках БД включить опцию «Включить удалённый обмен», а в поле IP-адреса ввести значение 192.168.56.101 в соответствии с адресом сервера.
- Значение поля «Порт удалённого сервера обмена» должно быть таким же, какое ранее было установлено для сервера, то есть 19000. Порт приёма данных может быть любым. Укажем для первого клиента порт приёма данных –19002, а для второго – 19003.
- В данном примере это не критично, так как у каждого из клиентов открыто всего одному порту, и они могут быть любыми, в том числе и 19000, как у сервера.

- У каждого из клиентов также надо включить опцию «Разрешить приём данных от клиентов» ввиду того, что сервер будет обратно присылать на клиенты данные, вычисляемые как на сервере, так и на других клиентах.

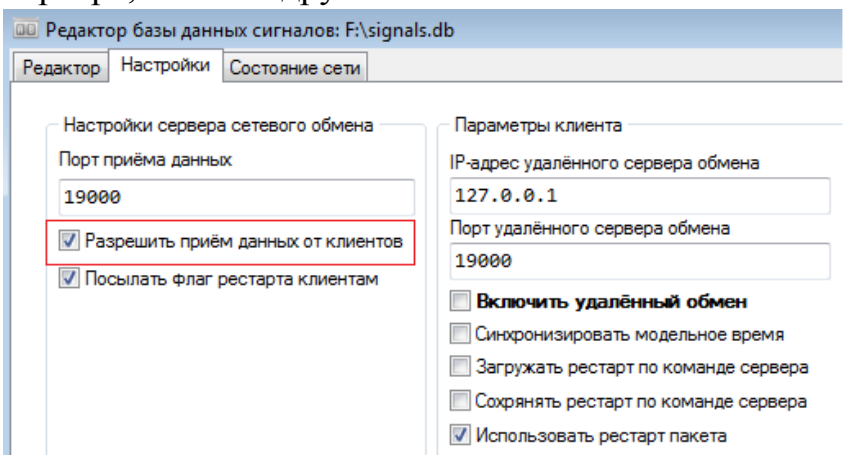

Рис. 6.22. Настройка сервера сетевого обмена.

- Важно у каждого из клиентов установить опцию «Синхронизировать модельное время» для того, чтобы синхронизировать модельное время расчета сервера и всех клиентских проектов. Общие настройки клиентов приведены на рис. 6.23.
-

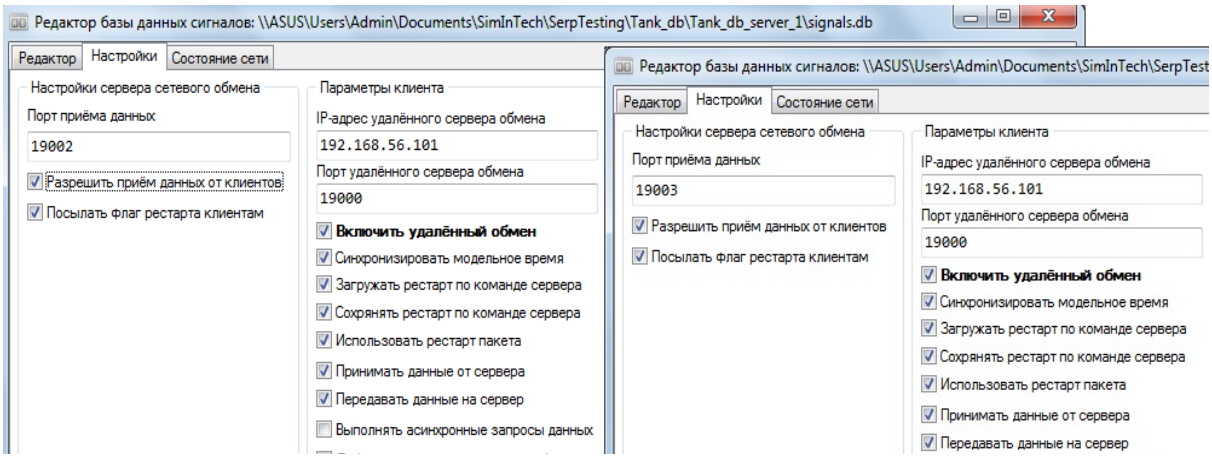

Рис. 6.23. Настройка БД в проекта tank_db_control и tank_db_object.

После выполнения всех этих настроек можно переходить к проверке работоспособности распределенной сетевой модели. Для этого надо запустить виртуальные машины и открыть в каждой из них свой проект, используя для загрузки проекта папку \\VBOXSVR\server\ или соответствующий сетевой диск. После того, как на каждом из узлов сети будет от-

крыт свой проект, надо последовательно, начиная с основного проекта, выполнить их инициализацию.

Отметим, что при этом в каждом из проектов расчетное время установится в 0, а статус перейдет в состояние ”Старт” (рис. 6.24). Если в окне основного проекта нажать кнопку ”Пуск”, то начнется распределённый расчет с синхронизацией модельного времени и обменом данными из базы данных сигналов. Таким образом, распределенная модель работает на всех узлах сети. Для остановки ее работы надо в окне основного проекта нажать кнопку ”Стоп” (рис. 6.25).

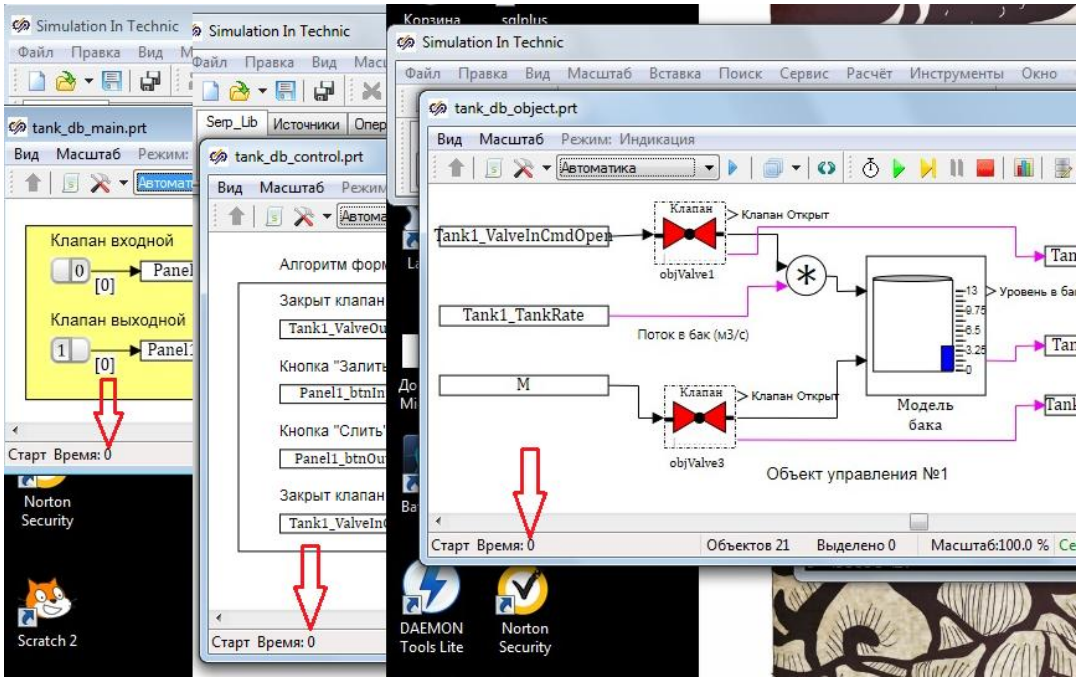

Рис. 6.24. Начальная инициализация проектов на каждом из узлов сети

.

Так как все проекты уже отлажены, и требуется исследование работы распределенной модели, то в этом случае загрузка основного меню среды разработки SimInTech становится ненужной. Ускорить работу пользователя по загрузке проектов распределенной модели можно, если после запуска виртуальной машины в ее командой строке ввести команду:

```
"C:\SimInTech\bin\mstarter.exe"/nomainform/exitonclose
<путь_до_проекта>/start
```

где <путь_до_проекта> на разных узлах будет иметь разные значения:

- \\VBOXSVR\server\tank_db_main.prt
- \\VBOXSVR\server\tank_db_control.prt
- \\VBOXSVR\server\tank_db_object.prt

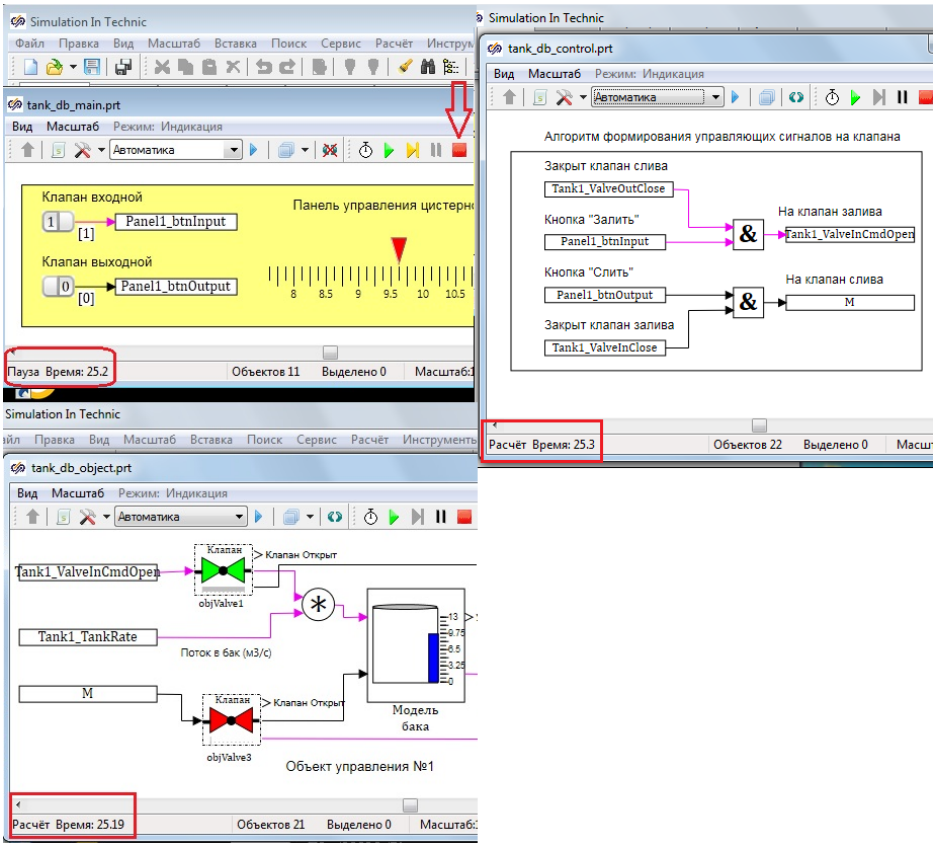

Рис. 6.25. Остановка распределенной системы.

Выполнение этой команды (рис. 6.26) обеспечит загрузку среды SimInTech, открытие нужного проекта и его автоматическую инициализацию, а также скрытие основного меню SimInTech. Кроме того, при закрытии окна проекта будет выполнена автоматическая выгрузка среды SimInTech из памяти компьютера.

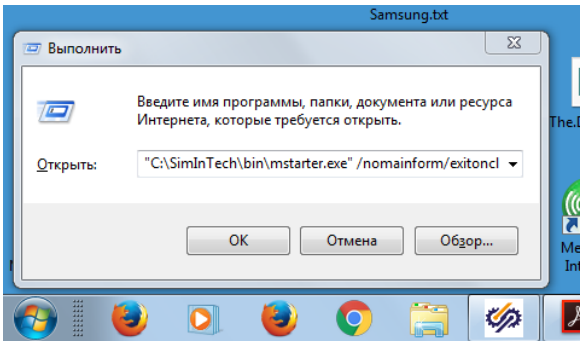

Рис. 6.26. Запуск клиента распределенной модели в работу.

Для удобства дальнейшей работы предпочтительнее создать ярлык, чтобы избавиться от необходимости многократного ввода этой команды при каждом запуске проекта. Можно еще более автоматизировать работу, если использовать эту команду в автозагрузке сетевого узла. Тогда при запуске виртуальной машины сразу откроется окно нужного проекта.

Подводя итог этому разделу, можно сделать вывод, что одним из способов реализации распределенных моделей является режим, так называемого, *сетевого расчёта в ручном режиме*. Для организации работы модели в этом режиме необходимо было выполнить следующие шаги.

- Настроить обмен данными через сеть и синхронизацию модельного времени, используя для этого возможности базы данных сигналов SDB.
- Инициализировать главный проект с включенным сервером сетевого обмена.
- Инициализировать всех клиентов − проекты с включенным удалённым обменом.
- Нажать кнопку ”Пуск” в окне главного проекта. При этом все проекты на разных вычислительных узлах запустятся на расчёт.

Следует отметить, что это не единственный режим реализации распределенных моделей, который используется в SimInTech. В частности, есть еще режим запуска сетевого расчёта в автоматизированном режиме (см. Приложение 6.2).

6.5. Задание на лабораторную работу

Если все рекомендации, изложенные в текущем разделе, выполнены, то в результате имеется отлаженная комплексная модель на базе пакета проектов (папка tank_db_1) и работоспособная распределенная модель (папка server).

Задания, которые предстоит выполнить, потребуют модификации и изменения в каждой из этих моделей. При этом работа с пакетом проектов по расширению его функциональности способствует получению более глубоких навыков работы с общесистемной базой данных, а задания по изменению распределенной модели должны дать представление об имеющихся в SimInTech более продвинутых способах запуска сетевого расчета. Исходя из этого подхода к изучению комплексных и распределенных моделей, в рамках данной лабораторной работы необходимо выполнить две группы заданий.

Задания по работе с пакетом проектов и его модификации.

1. Скопировать папку tank_db_1 в папку tank_db_2 , переименовать файл пакета в tank_db_pak2.pak и дальше работать с этим пакетом.

2. Расширить функциональность проекта, который теперь должен работать сразу с двумя объектами управления. Для этого в проекте tank_db_object.prt скопировать полностью фрагмент блоков "Объект управления №1" в "Объект управления №2" и изменить в нем привязку к сигналам базы банных (рис. 6.27).

3. Аналогично изменить и другие проекты, сохранить сделанные изменения. После чего загрузить пакет tank_db_pak2.pak, запустить его на выполнение и протестировать работу комплексной модели, подавая сигналы на клапаны обеих цистерн (рис. 6.28).

4. Познакомиться с возможностями векторной обработки сигналов (см. Приложение 6.1 и справку SimInTech). Изменить проект tank_db_control.prt таким образом, чтобы в нем использовалась бы только одна версия алгоритма, который может работать с любым числом панелей и объектов управления, прописанных в базе данных. Выполнить тестирование измененного проекта.

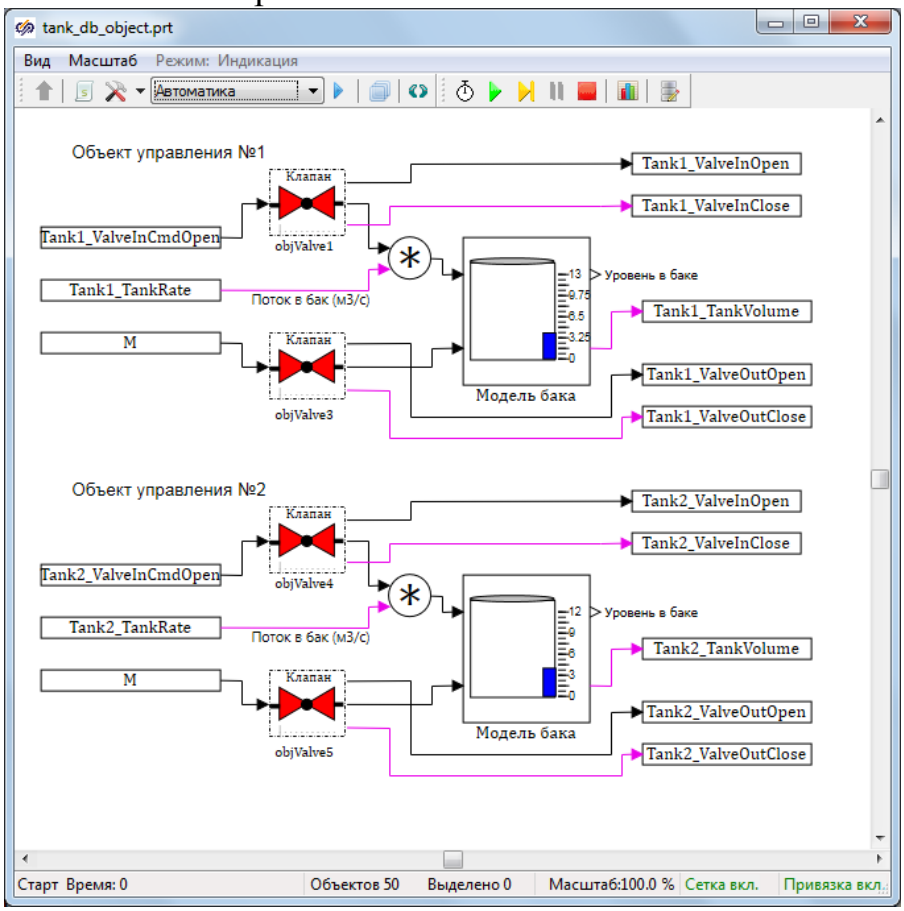

Рис. 6.27. Модификация проекта tank_db_object.prt

5. Добавить в базу данных еще одну группу сигналов, например, Tank3 и убедится в том, что при запуске пакета проектов на выполнение появляется ошибка при загрузке проекта tank_db_control.prt.

6. Для того, чтобы выполнялась проверка возможности работы комплексной модели при текущем состоянии базы данных, а также выводилось понятное пользователю сообщения о несогласованности базы данных, необходимо проект tank_db_main.prt дополнить следующим скриптом (рис. 6.29).

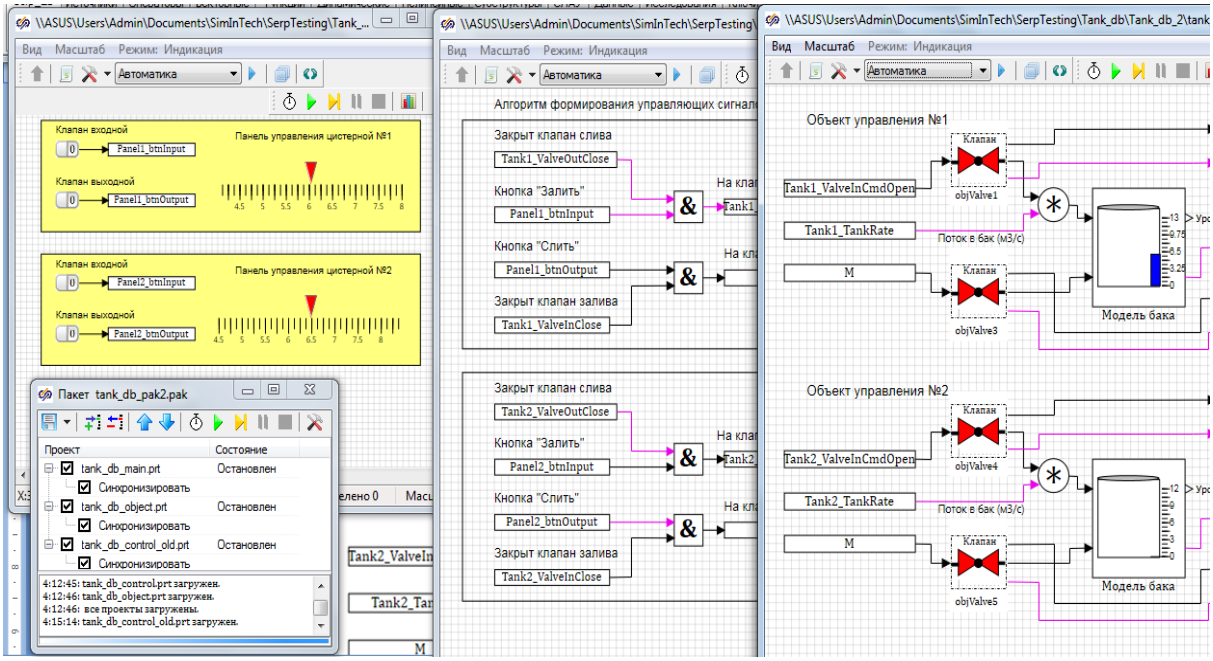

Рис. 6.28. Тестирование работы комплексной модели.

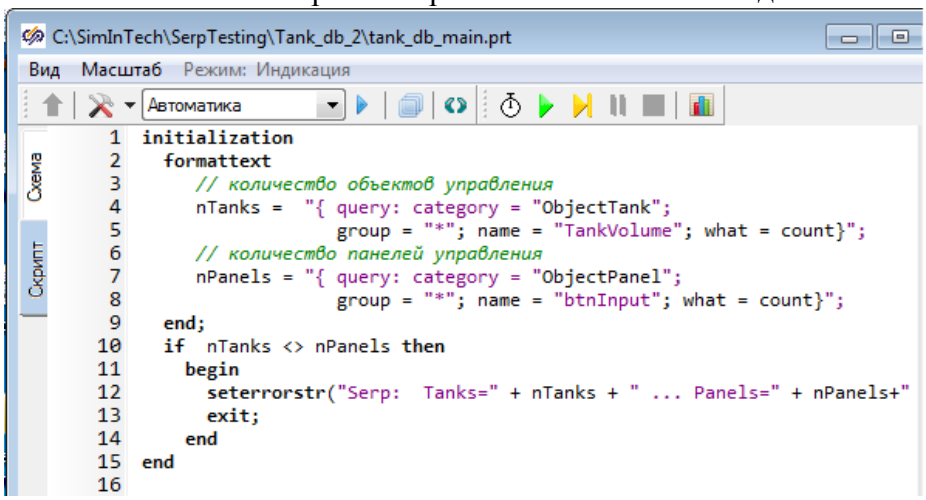

Рис. 6.29. Скрипт проверки несогласованности базы данных.

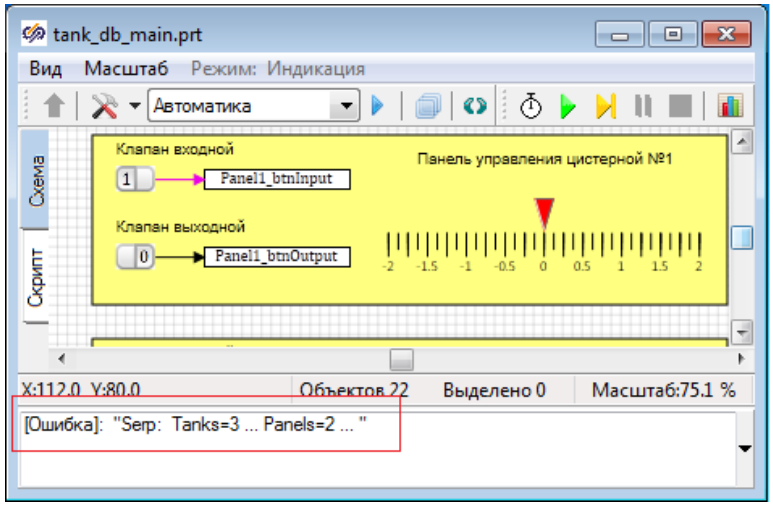

Рис. 6.30. Работа пакета проектов при несогласованной базе данных.

7. Проверить работу tank_db_main.prt при запуске пакета на выполнение (рис. 6.30).

8. Привести базу данных в исходное состояние и подготовить пакет проектов для его преобразования в распределенную сетевую модель.

Задания по работе с распределенной моделью и ее сетевому тестированию:

1. Выполнить тестирование новой распределенной модели в режиме сетевого расчёта в ручном режиме.

2. Используя среду виртуализации, настроить распределенную модель для запуска сетевого расчёта в автоматизированном режиме.

3. Используя сегмент реальной сети, выполнить настройку отдельных ее узлов, а также самой распределенной модели для возможности ее работы на отдельных сетевых станциях.

Три дополнительных задания:

1. Предусмотреть на панелях управления индикацию о достижении аварийно высоких и аварийно низких уровнях жидкости в цистернах.

2. Создать новый пакет проектов, который бы использовал расширенную базу данных, которая позволяла бы описывать все параметры модели, включая геометрию каждой из цистерн, индивидуальное время срабатывания каждого из клапанов и т.д.

3. Разработать пакет проектов, который объединил бы в себе возможности ручного и автоматического управления заливкой цистерн.

Приложение 6.1. Векторная обработка сигналов

При моделировании сложных систем очень часто возникает ситуация, когда необходимо многократно использовать одну и ту же типовую математическую модель. Среда SimInTech позволяет использовать одну математическую модель для нескольких однотипных объектов с помощью механизма векторной обработки сигналов.

В этом разделе рассматриваются некоторые аспекты векторной обработки сигналов на примере разработанной выше комплексной модели, в которой отдельные проекты связаны между собой общесистемной базой данных. Поставим задачу разработки и включения в состав пакета типового блока управления клапанами.

С этой целью в проекте tank_db_control.prt из пакета tank_db_pak2.pak оставим только одну версию алгоритма управления и изменим привязку его входных и выходных блоков к базе данных. Этот алгоритм должен быть общим для всех клапанов модели, поэтому все его внешние сигналы будут векторными. При этом каждый блок чтения или записи сигналов в список должен обрабатывать столько сигналов, сколько клапанов и объектов управления существует в базе данных.

Что же привлекательного в этом подходе? Прежде всего то, что при следующем запуске этой распределенной системы этот проект можно запустить уже не в демонстрационном зале, а на компьютере разработчика или инженера со всеми открытыми меню и опциями SimInTech. А все что нужно сделать для этого, это в утилите массового запуска изменить в одной строке вкладки «Экраны» два параметра: номер строки шаблона и IP-адрес компьютера, где запускается проект.

Заключение

Среда динамического моделирования технических систем SimInTech позволяет на основе дружественного простого интерфейса создавать сложные распределенные проекты, использующие общую базу данных сигналов и обладающие богатыми средствами визуализации происходящих в работающей модели процессов, в том числе средствами анимации.

К сожалению, в силу сложившихся обстоятельств в отечественной высшей школе в настоящее время доминируют преимущественно зарубежные программы, в частности, при изучении дисциплин, связанных с автоматическим управлением, распределенной обработкой, используется в большинстве случаев зарубежное программное обеспечение, например, Simulink (MathLab), для которых выпущено большое количество электронных пособий и печатных учебных изданий. Применение зарубежных программных средств, при наличии не менее высокоэффективных отечественных программных продуктов, противоречит наметившейся в последнее время государственной экономической политике импортозамещения в различных отраслях народно-хозяйственной деятельности РФ, в том числе и в области информационных технологий в части программного обеспечения. SimInTech − среда *визуального* проектирования и программирования с русскоязычной обширной справочной системой и удобным для отечественных пользователей интерфейсом, не уступающая по своим функциональным возможностям и характеристикам зарубежным аналогам.

Визуальные, интерактивные средства программирования, используемые в SimInTech, сопроводительная документация отличаются "прозрачностью" и доступностью для понимания, позволяют обучающимся, при наличии элементарных навыков работы в ОС Windows, за короткий срок изучить среду SimInTech и успешно в ней работать. Следует отметить, что ПО SimInTech совместимо с ОС "Гослинкус", являющейся дистрибутивом сертифицированной в России свободно и открыто распространяемой ОС Linux, что немаловажно при необходимости перехода с ОС Windows на ОС Linux.

Для изучения методов математического моделирования автоматических систем и распределенной обработки данных в этих системах, преду-

смотренных рамками настоящего учебного пособия, достаточно знать основные принципы работы систем автоматизированного регулирования, а также иметь навыки работы в среде ОС Windows. В результате последовательного выполнения рекомендаций данного пособия, обучающиеся построили проект распределенной обработки, в котором применены средства анимированной визуализации, с возможностью изменять в ходе выполнения приложения значения основных его параметров, причем делать это независимо друг от друга на разных сетевых станциях.

Список литературы

1. Карташов Б.А., Шабаев Е.А., Козлов О.С., Щекатуров А.М.. Среда динамического моделирования технических систем SimInTech, − Москва: ДМК Пресс, 2017. – 423 с.

2. Баум Ф.И. Программное обеспечение SimInTech для программирования приборов систем управления / Ф.И. Баум, О.С. Козлов, И.А.Паршиков, В.Н. Петухов, К.А.Тимофеев, А.М. Щекатуров // Атомная энергия, 2012. Т.113. Вып.6 С.354-357.

3. Щекатуров А.М. Методика разработки функционального программного обеспечения АСУ ТП ЯЭУ с применением модельно-ориентированного подхода в SimInTech / А.М. Щекатуров, К.А. Тимофеев, О.С. Козлов // Университетский научный журнал. 2015. №15. С.80-87.

4. Справочная система SimInTech. 2017. [Электронный ресурс]. URL: http://simintech.ru/help/rus/.

Оглавление

ВВЕДЕНИЕ 3
1. ОСНОВЫ РАБОТЫ В СРЕДЕ SIMINTECH 6
1.1. Концепция создания комплексной модели объекта в среде SimInTech 6
1.2. Первый запуск среды SimInTech 7
1.3. Задание на лабораторную работу 12
2. РАЗРАБОТКА ПРОСТЕЙШИХ МОДЕЛЕЙ 13
2.1. Создание нового проекта схемы автоматики 14
2.2. Начало создания модели системы управления клапаном 16
2.3. Модификация модели системы управления клапаном 20
2.4. Генерация кода на языке Си и внешней dll-библиотеки для алгоритма управления клапаном 22
2.5. Исследование работы клапана с алгоритмом управления из подключаемой внешней dll-библиотеки 24
2.6. Коротко о кодогенерации 26
3. ПОСТРОЕНИЕ МОДЕЛИ УПРАВЛЕНИЯ ЗАЛИВКОЙ ЦИСТЕРН 27
3.1. Разработка модели заполнения и осушения цистерны 28
3.2. Разработка алгоритма управления заполнением и осушением цистерны 32
3.3. Задание на лабораторную работу 38
4. ОСНОВЫ АНИМАЦИИ БЛОКОВ И СОЗДАНИЯ СОБСТВЕННЫХ БИБЛИОТЕК. 40
4.1. Создание анимированного изображения для блока цистерн 42
4.2. Подготовка блока для его включения в библиотеку 48
4.3. Создание собственной библиотеки блоков 53
4.4. Задание на лабораторную работу 56
Приложение 4.1. Пример разработки библиотечных блоков, моделирующих работу клапанов 56
Блок субмодели 58
Формирование графического контейнера для блока клапана 59
Модификация графического контейнера блока клапана 61
5. ПРОСТЕЙШИЙ ПОДХОД К ОРГАНИЗАЦИИ РАСПРЕДЕЛЕННОГО СЕТЕВОГО РАСЧЕТА 63
5.1. Пример использования связи по UDP-протоколу 64
5.2. Использование UDP для построения распределенной модели управления цистерной 69
5.3. Декомпозиция модели для ее распределенной работы 71
5.4. Задание на лабораторную работу 75
Приложение 5.1. Настройка виртуальных машин для тестирования распределенных проектов 79
Приложение 5.2. Основные свойства UDP блоков SimInTech 82
Блок UDP-клиента 82
Блок UDP-сервер 83

Блок «Мультиплексор» 83
Блок «Демультиплексор» 84
6. ОРГАНИЗАЦИЯ РАСПРЕДЕЛЕННЫХ МОДЕЛЕЙ С ОБЩЕСИСТЕМНОЙ БАЗОЙ ДАННЫХ 84
6.1. Формирование модели ручного управления клапанами 86
6.2. Подключение базы данных к пакету проектов 88
6.3. Заполнение базы данных и связь блоков модели с сигналами базы данных 92
6.4. Реализация распределенной модели с общей базой данных 98
6.5. Задание на лабораторную работу 105
Приложение 6.1. Векторная обработка сигналов 108
Приложение 6.2. Настройка запуска сетевого расчета в автоматизированном режиме .112
Заключение 119
Список литературы 120

С о с т а в и т е л и :
Хабаров Сергей Петрович
Шилкина Мария Львовна

РАСПРЕДЕЛЕННАЯ ОБРАБОТКА ДАННЫХ

* * *

П О С Т Р О Е Н И Е Р А С П Р Е Д Е Л Е Н Н Ы Х М О Д Е Л Е Й В С И С Т Е М Е S I M I N T E C H

Методические указания
по выполнению лабораторных работ
для бакалавров направления 09.03.02
"Информационные системы и технологии"

Отпечатано в авторской редакции с готового оригинал-макета

Подписано в печать с оригинал-макета 29.03.18.
Формат 60 × 84/16. Бумага офсетная. Печать трафаретовая.
Уч.-изд.л.7,75 Печ.л.7,75 Тираж 125 экз. Заказ №35. С18.

Санкт-Петербургский государственный лесотехнический университет
Отпечатано с готового оригинал-макета, предоставленного автором,
в типографии ООО ГАЛАНИКА
Санкт-Петербург, ул. Правды, 15.