**ЛАБОРАТОРНАЯ РАБОТА № 3**

**АНАЛИЗ В СРЕДЕ SIMINTECH ДИНАМИЧЕСКИХ СИСТЕМ,**

**ЗАДАННЫХ В ФОРМЕ КОШИ И В ПЕРЕМЕННЫХ СОСТОЯНИЯ**

# ЦЕЛЬ РАБОТЫ

* ознакомление с методами формирования математических моделей динамики САР в среде SimInTech, включая:
  + процедуры создания многоуровневых структурных схем;
  + механизм **Скрипта** (в части задания глобальных констант и переменных);
  + описание математической модели динамики САР в переменных состояния;
  + использование встроенного **Языка программирования** (и его Интерпретатора математических функций) для задания уравнений динамики САР;
  + методы реализации "беспроводной" передачи данных;
* самостоятельное исследование переходных процессов в простейшей модели САР ядерного реактора (ЯР), включая:
  + приведение математического описания динамики САР ЯР к задаче Коши;
  + формирование математической модели САР ЯР с использованием:
    - описания в переменных состояния (в матричной форме);
    - блока "Язык программирования", который позволяет качественно расширить варианты компактного преобразования структурной схемы САР;
* моделирование переходных процессов в САР ЯР при подаче управляющих и возмущающих воздействий.

# ВВЕДЕНИЕ

В лабораторных работах №1 и №2 Вами рассмотрены процедуры работы в среде SimInTech применительно к анализу динамических процессов в САР, модель динамики которой описывалась в переменных "вход-выход". Но – такой вид задания структурной схемы САР вполне приемлем только для несложных задач учебного плана.

Во многих учебных, и тем более в отраслевых задачах, дифференциальные уравнения динамики объекта исследования записываются в форме Коши, а именно: как система нелинейных дифференциальных уравнений 1-го порядка, разрешенных относительно производных. Попытка реализовать в переменных "вход-выход" даже простейшие нелинейные операции в правых частях уравнений, используя для этого соответствующие типовые блоки (⨯, /, **sin**, **ln** и т.п.), приводит к резкому увеличению размера структурной схемы. Поэтому формирование структурной схемы динамики САР с использованием только относительно простых типовых блоков является неэффективным: структурная схема становится "нечитаемой", так как, зачастую, не может быть полностью отображена в размерах экрана одного дисплея.

При формировании относительно сложной структурной схемы САР (или другой динамической системы) целесообразно мысленно (или на листе бумаги!) «расчленить» структурную схему на несколько отдельных (относительно не громоздких) фрагментов. Далее – сформировать структурную схему каждого фрагмента в виде отдельной **Субмодели**, имеющей входы и выходы (входные и выходные порты, соответственно), и затем набрать структурную схему всей системы из **Субмоделей**.

Такой подход используется во многих программных комплексах, реализующих методы структурного моделирования, в том числе и в среде SimInTech. Однако в среде SimInTech реализован ряд методических решений, позволяющих качественно сократить количество блоков в структурной схеме (вплоть до порядков).

Перейдем к "знакомству" с указанными методическими решениями.

# 1 МЕТОДЫ ФОРМИРОВАНИЯ МАТЕМАТИЧЕСКОЙ МОДЕЛИ САР В SIMINTECH

## 1.1 Преобразование математического описания САР и формулирование задач очередного дополнительного задания

В данном разделе будут рассмотрены наиболее значимые программно-методические решения, реализованные в среде SimInTech, не имеющие аналогов в отечественных и в большинстве известных зарубежных программных комплексах подобной направленности.

Программные возможности среды SimInTech для формирования математической модели динамики объекта исследования рассмотрим в процессе выполнения нового многоэтапного дополнительного задания к демонстрационно-ознакомительной задаче, структурная схема которой после выполнения заданий по расчету амплитудно-фазовых частотных характеристик САР имела вид, близкий рисунку 1.1.



Рис. 1.1 – Исходная САР

В рассматриваемой демонстрационно-ознакомительной задаче Объект Управления описывается тремя блоками (звеньями), а именно: блоки с передаточными функциями **W­₂(s)** и **W₃(s)**, а также блок **Локальное сравнивающее устройство**.

«Легко видеть», что динамика Объекта Управления в данной задаче описывается следующей системой линейных уравнений:

|  |  |
| --- | --- |
|  | (1.1) |

при нулевых начальных условиях, т.е.

Введя новые динамические переменные и можно привести математическое описание динамики Объекта Управления к форме Коши:

|  |  |
| --- | --- |
|  | (1.2) |

где

а начальные условия для новых динамических переменных равны нулю.

К форме Коши можно привести математическое описание уравнений динамики САР в целом, добавив к системе (1.2) уравнение динамики для интегрирующего регулятора (блок с подписью **W₁(s)**) и алгебраическое соотношение для **Главного сравнивающего устройства**. В этом случае система уравнений динамики САР, "подготовленная" для перехода к описанию в переменных состояния, примет вид:

|  |  |
| --- | --- |
|  | (1.3) |

где

После ввода очередных новых динамических переменных , равных, соответственно, , описание динамики САР можно представить в матричной форме, а именно, в переменных состояния:

|  |  |
| --- | --- |
|  | (1.4) |

где вектор-столбец производных переменных состояния *x'(t)*, вектор-столбец переменных состояния *x(t)*, матрица системы *А*, матрица входа *В*, матрица выхода *С* и матрица обхода *D* равны (в этой задаче):

|  |  |
| --- | --- |
|  | (1.5) |

Сформулируем, наконец, отдельные задачи, которые Вам предстоит решить при выполнении данного *очередного дополнительного* задания:

1. Добавить к структурной схеме "основной" САР (см. рис. 1.1) новую "параллельную" САР в виде **Субмодели**.
2. Задать коэффициенты системы уравнений "параллельной" САР через **Скрипт** и глобальные константы (или не меняющиеся в процессе счета глобальные переменные, другое название которых – параметры САР).
3. Задать описание уравнений динамики Объекта Управления (см. систему (1.2)) в "параллельной" САР с использованием блока **"Язык программирования"** из библиотеки **Динамические** звенья.
4. Задать описание уравнений динамики САР в целом (см. системы (1.3) и (1.4)) с использованием типового блока **Переменные состояния** из библиотеки **Динамические** звенья, расположенного внутри "параллельной" САР.
5. Реализовать "беспроводной" обмен данными между "основной" и "параллельной" САР, используя блоки **В память** и **Из памяти** из библиотеки **Субструктуры**.

Следует заметить, что предложенные Вам в качестве этапов очередного дополнительного задания пункты 2, 3 не могут быть выполнены в среде других программных комплексов аналогичной направленности (ни в отечественных, ни в зарубежных), а пункты 1 и 4 в полном объеме могут быть выполнены только в среде наиболее популярного зарубежного программного комплекса – в SimuLink.

## 1.2 Создание "параллельной" САР в виде новой Субмодели

Добавление к структурной схеме (см. рис. 1.1) "параллельной" САР в виде новой Субмодели выполним в следующей последовательности:

**Этап 1** – перенос типового блока **Субмодель** в Схемное Окно.

Учитывая, что проект (задача) со структурной схемой, аналогичной рис. 1.1, была сохранена Вами на жесткий диск, откройте его (проект). Переместите курсор на закладку Субструктуры в "Линейке" типовых блоков и инициализируйте ее щелчком левой клавиши "мыши". Далее из "Линейки" типовых блоков перенесите в Схемное Окно блок **Субмодель** таким же образом, как ранее Вы это делали с другими блоками при выполнении демонстрационно-ознакомительной задачи. В поле Схемного Окна появится новый блок – **Субмодель** (без входных и выходных портов).

**Этап 2** – заполнение внутренней структуры **Субмодели**.

Переместите в Схемном Окне курсор на блок Субмодель и выполните 2-х кратный щелчок левой клавишей мыши: произойдет открытие субмодельного схемного окна (т.е. переход на 1-ый уровень вложенности).

Перенесите в открывшееся чистое схемное окно (субмодельное) из этой же библиотеки (Субструктуры) блок **Порт входа** (желательно в левую часть схемного окна) и блок **Порт выхода** (лучше в правую часть схемного окна), а также остальные блоки: *Сравнивающее устройство*, *Интегратор*, *Временной график* и блок *Язык программирования*.

Хотя на данном этапе выполнения задания в субмодельном окне невозможно провести все линии связи между блоками (так как блок *Язык Программирования* не имеет пока ни входного, ни выходного портов), проведите в субмодельном схемном окне те линии связи между блоками, которые можно провести. После оформления поясняющих подписей субмодельное схемное окно будет иметь вид, подобный рис. 1.2.



Рис. 1.2 – «Параллельная» САР

Закройте субмодельное схемное окно 2-х кратным щелчком левой клавишей "мыши" в свободном месте субмодельного схемного окна. Изображение **Субмодели** в Главном Схемном Окне изменится: появятся входной и выходной порты.

Если условия какой-то решаемой Вами задачи требуют, чтобы субмодельная структура имела, например, 3 входных порта и 2 выходных порта, то Вы должны "перенести" в субмодельное схемное окно три **Порта входа** и два **Порта выхода**. Первому перенесенному в это окно **Порту входа** будет соответствовать верхний входной порт (при ориентации блока **Субмодель** слева-направо), второму – средний входной порт, а третьему – нижний входной порт (при ориентации блока **Субмодель** слева-направо). Аналогично, первому перенесенному в субмодельное окно Порту выхода будет соответствовать верхний выходной порт (при ориентации блока **Субмодель** слева-направо) и т.д. Примечание: рекомендуется именовать порты входа и выхода субмодели уникальным образом (для данной субмодели), во избежание путаницы!

Входные порты **Субмодели** могут быть векторными (многожильными), например, по первому входу передается одновременно 5 сигналов, по второму – 6 сигналов, а по третьему – 4 сигнала. Очевидно, что подобные векторные сигналы должны быть предварительно сформированы, например, посредством блока **Мультиплексор**. Внутри субмодельного схемного окна выходные сигналы блоков **Порт входа**, обычно, должны быть демультиплексированы для последующей обработки.

Векторными могут быть и выходные порты субмодели, что требует соответствующего предварительного мультиплексирования выходных сигналов, подаваемых внутри субмодели на блоки **Порт выхода**.

Внимание! Если в субмодели несколько **Портов входа** и **Портов выхода**, рекомендуется сразу после переноса каждого Порта давать ему собственное имя. Например, дать 1-му перенесенному **Порту входа** "уникальное" имя: Вход №1 и т.д.

**Этап 3** – включение **Субмодели** в основную структурную схему.

Соедините **Субмодель** линиями связи с блоком **Управляющее воздействие** и блоком **График y(t)**, предварительно изменив на 2 число входов в блоке **Временной график**. Сделайте поясняющую подпись у нового блока **Субмодель**. Главное Схемное Окно примет вид, подобный рис. 1.3.

На этом процесс включения созданной Вами новой субмодели ("Параллельная" САР) в основную структурную схему САР почти завершен.



Рис. 1.3

Аналогичным способом можно внести в Главное Схемное Окно еще несколько новых субмоделей, а также можно внести новые субмодели и в субмодельное схемное окно (субмодели 2-го уровня вложенности).

При щелчке правой клавишей "мыши" по блоку **Субмодель** появляется контекстное меню, которое имеет дополнительную опцию: *Действия*→*Сохранить и связать с файлом*, которая позволяет сохранить Субмодель на жестком диске под оригинальным именем (в отдельном файле проекта с расширением .prt) и использовать ее как "заготовку" при формировании структурной схемы другой задачи.

Диалоговое окно свойств **Субмодели** (см. рис. 1.4), вызываемое щелчком правой клавиши "мыши" по блоку, и затем щелчком левой клавишей "мыши" по опции **Свойства**, имеет в закладке «Общие» среди прочих две часто нужные диалоговые строки:

* Графическое изображение – предназначена для изменения пиктограммы блока на новую (в bmp-формате);
* Цвет фона субмодели – предназначена для замены однотонного фона субмодельного окна (по умолчанию белого) на рисунок (в bmp-формате), например, с изображением мнемосхемы, на фоне которой создается структурная схема в данном субмодельном окне.

Необходимо заметить, что изменение фона субмодельного окна требует от Пользователя предварительного "конструирования" образа получаемой схемы и рисунка.

Задание **Скрипта** для каждой субмодели можно (иногда это необходимо) осуществлять, зайдя внутрь неё и перейдя на вкладку **Скрипт**, аналогично вкладке **Скрипт** на Главном Схемном Окне.



Рис. 1.4 – Свойства Субмодели

## 1.3 Задание параметров САР через механизм Скрипта

Встроенный в SimInTech Интерпретатор математических функций (Язык Программирования в блоке с одноименным названием) обеспечивает функционирование окна вкладки **Скрипт** Проекта (Субмодели), посредством которого можно в т.ч. реализовать задание свойств блоков структурной схемы, определив константы и переменные Проекта (Субмодели).

Окно Редактора **Скрипта** Проекта (Субмодели) – фактически окно текстового редактора, в котором Вы можете, по определённым правилам, задать значения или выражения ряда параметров САР, называемых Глобальными переменными (константами), которые в процессе моделирования остаются постоянными либо изменяющимися по какому-то закону, описанному в скрипте. Интерпретатор математических функций "распознает" более 30-ти операторов, включая чисто математические (+, -, \*, /, **sin**, **tg**, **ln** и т.п.), логические (**if**, **for** и др.), ключевые слова и системные константы (**time**, **step**, **goodstep**, **pi** и др.), функциональные операторы (**interpol**, **fft** и др.), а также более 200-х различных функций. Более подробная информация об Интерпретаторе математических функций будет представлена в следующем подразделе (также см. справку).

Откройте окно редактора **Скрипта** Проекта, расположенной слева на Схемном окне.

Введите текст, описывающий задание глобальных констант, и комментарии (в фигурных скобках {…}, можно также использовать двойной слэш // для однострочных комментариев), аналогично рис. 1.5.

После ввода основного текста (цвет шрифта – черный и синий) и комментариев к нему (цвет шрифта – зеленый), перейдите обратно на вкладку **Схема**, т.е. закройте окно редактора Скрипта Проекта (Субмодели). Для этого надо щелкнуть "мышью" по вкладке **Схема** (верхняя слева над вкладкой «Скрипт»), либо нажать кнопку Применить (зелёная галочка в правой части окна).

На этом процедуры задания Глобальных констант завершены.

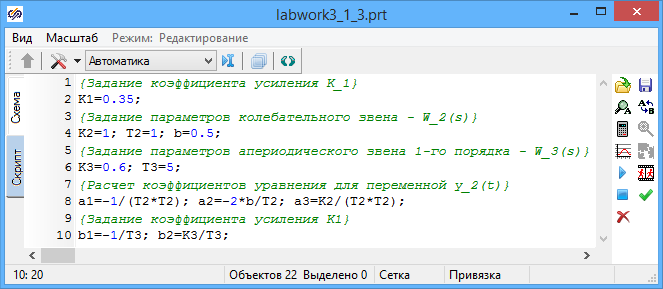


Рис.1.5

Если в проекте (задаче) какие-то параметры САР заданы как глобальные константы в Главном Схемном Окне, то они могут быть использованы для задания свойств конкретных блоков не только в этом окне, но и во всех вложенных структурах (Субмоделях).

**Примечание 1**: представленным на рисунке 1.5 скриптом вы определили глобальные переменные проекта; но, в силу того что никто эти переменные не переопределет (ни в какой другой части проекта эти переменные не стоят в левых частях), они будут оставаться постоянными. Для определения констант следует воспользоваться ключевым словом **const**.

**Примечание 2**: строго говоря, каждая строка представленного на рис. 1.5. скрипта будет исполняться на каждом расчетном шаге; для исключения этих излишних действий необходимо заключить все 10 строк в секцию **initialization** … **end**; что будет означать инструкцию исполнить эти строки скрипта только на этапе инициализации схемы. Чтобы не усложнять лабораторную работу, здесь мы этого делать не будем, но в реальных проектах так надо поступать в обязательном порядке, для ускорения расчета, и устранения нежелательных действий в процессе счёта.

Если какая-то глобальная переменная/константа (заданная в скрипте Главного Схемного Окна) повторно задана во вложенной субмодели, то последняя переопределяет значение ранее заданной переменной и её можно использовать для задания свойств конкретных блоков как в данном субмодельном схемном окне, так и во всех "дочерних" субмоделях более глубокого уровня вложенности. **Примечание**: во избежание путаницы так лучше не делать, однако в больших проектах это бывает необходимо.

## 1.4 Формирование уравнений динамики с использованием блока «Язык программирования»

Учитывая, что невозможно сформировать абсолютно полную библиотеку типовых блоков, в среде SimInTech разработаны средства, которые позволяют Пользователю расширить состав личной библиотеки за счет создания новых типов блоков, например, посредством встроенного в SimInTech языка программирования.

На базе языка программирования функционирует один из "нестандартных" типовых блоков библиотеки **Динамические** звенья, а именно, блок **Язык программирования**, позволяющий прямо в процессе работы создавать экземпляры блоков со своими оригинальными математическими моделями.

Диалоговое окно этого блока – полностью аналогично окну редактора скрипта Проекта. Пользователь в текстовом виде записывает алгебраические уравнения и уравнения динамики в виде, близком к их записи ручкой на листе бумаги.

Математическое описание блока может соответствовать многомерной нелинейной динамической системе в форме Коши:

|  |  |
| --- | --- |
|  | (1.6) |

где – известные нелинейные функции переменных и входных воздействий , причем в качестве входных воздействий могут выступать и коэффициенты (постоянные или переменные), входящие в любое из уравнений (1.6).

Первое из уравнений системы (1.6) может отсутствовать: в этом случае блок **Язык программирования** выполняет алгебраические преобразования входных величин. Использование этого блока в качестве функционального весьма эффективно в случае наличия в модели сложных функциональных преобразований, когда использование для этих целей элементарных типовых блоков приведет к неоправданному усложнению структурной схемы.

Блок **Язык программирования** "распознает", в частности, следующие стандартные математические операции и функции:

|  |  |
| --- | --- |
| **+** | – сложение; |
| **abs** | – модуль; |
| **sin** | – синус; |
| **arcsin** | – арксинус; |
| **-** | – вычитание; |
| **sign** | – знак; |
| **cos** | – косинус; |
| **arccos** | – арккосинус; |
| **\*** | – умножение; |
| **exp** | – экспонента; |
| **tg** | – тангенс; |
| **arctg** | – арктангенс; |
| **/** | – деление; |
| **ln** | – логарифм; |
| **ctg** | – котангенс; |
| **arcctg** | – арккотангенс; |
| **^** | – степень; |
| **pi** | – 3.1415…; |
| **e** | – 2.7182818284590452353602874713527; |
| **( )** | – cкобки; |

Более подробно о языке программирования можно прочесть во встроенной справке SimInTech.

В язык программирования и, соответственно, в блок **Язык программирования** включено также 8 специальных функций, реализующих определение основных термодинамических свойств воды и водяного пара (известные таблицы Вуколовича в докритическом диапазоне: давления от 0.09 до 50 МПа и температуры от 10 до 800 °С). Запись этих функций имеет вид:

|  |  |
| --- | --- |
| Функция: | Вычисление свойств: |
| **waterps**(P, flag); | – воды по давлению на линии насыщения; |
| **waterts**(T, flag); | – воды по температуре на линии насыщения; |
| **steamps**(P, flag); | – пара по давлению на линии насыщения; |
| **steamts**(T, flag); | – пара по температуре на линии насыщения; |
| **waterpt**(P,T, flag); | – воды по давлению и температуре; |
| **waterph**(Р,H, flag); | – воды по давлению и энтальпии; |
| **steampt**(P,T, flag); | – пара по давлению и температуре; |
| **steamph**(P,H, flag); | – пара по давлению и энтальпии. |

Например, запись

v\_уд = **waterps**(1е6, 4);

соответствует вычислению удельного объема воды v\_уд (flag равен 4) на линии насыщения при давлении 1·10⁶ Па (1 Мпа).

Значения параметра flag могут изменяться от 1 до 12 и соответствовать вычислению следующих термодинамических характеристик воды или водяного пара:

1 – давление;

2 – температура;

3 – энтальпия;

4 – удельный объем;

5 – число Прандтля;

6 – динамическая вязкость;

7 – коэффициент теплопроводности;

8 – энтропия;

9 – удельная теплоемкость Ср;

10 – удельная теплоемкость Сv;

11 – производная плотности по энтальпии при постоянном давлении ;

12 – производная плотности по давлению при постоянном объеме ;

Входные параметры и возвращаемые значения термодинамических параметров воды или водяного пара представляются в системе СИ (за исключением температуры, которая измеряется в градусах Цельсия).

Освоение процедур формирования математической модели динамики какого-то устройства или фрагмента рабочего процесса с использованием блока **Язык программирования** рассмотрим на примере формирования уравнений динамики Объекта Управления, расположенного в субмодели 1-го уровня вложенности (см. рис. 1.2).

Откройте субмодельное окно "Параллельная" САР, переместите курсор на блок Объект Управления (см. рис. 1.2) и выполните 2-х кратный щелчок левой клавишей "мыши": откроется новое окно Редактора, в котором Вы должны записать выражения и дифференциальные уравнения, соответствующие математической модели данного блока (система уравнений (1.2)).

Панель инструментов окна редактора блока **Язык программирования** и его интерфейс полностью идентичны описанному выше окну редактора **Скрипта** Проекта (Субмодели).

Введите с клавиатуры (в текстовом виде) математическую модель динамики блока Объект Управления, как это показано на рис. 1.6, где представлена копия окна Редактора с математической моделью динамики рассматриваемого блока и комментариями к ней.

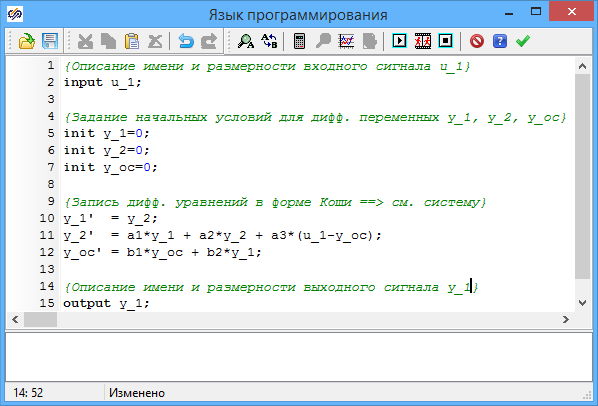


Рис. 1.6 – Математическая модель динамики рассматриваемого Объекта Управления

Если блок **Язык программирования** имеет входы (входные порты), то первая исполняемая строка (не считая строку комментария) обязательно должна содержать оператор **input**, описывающий входные сигналы в данный блок, включая имя входа и его размерность.

В данном примере 1-я исполняемая строка (**input** u\_1**;**) присваивает 1-му (и единственному) входу уникальное имя **u\_1**. Если бы, например, этот блок имел бы 2 входа, причем 1-ый вход – "трёхжильный" (векторный), а 2-ой – "пятижильный", то 1-я исполняемая строка имела бы вид:

**input** u1[3], g[5]**;**

Для описания размерности входов используются прямоугольные скобки.

Если блок **Язык программирования** описывает динамику объекта моделирования в виде системы дифференциальных уравнений в форме Коши, то вторая исполняемая строка (не считая строку комментария) обязательно должна содержать оператор **init**, описывающий начальные условия для динамических (дифференциальных) переменных, для которых ниже по тексту будут записаны обыкновенные дифференциальные уравнения в форме Коши.

В данном примере 5-я и последующие две исполняемые строки (**init** y\_1=0; **init** y\_2=0; **init** y\_oc=0;) задает начальные условия (при t = 0) для 3-х динамических переменных: *y\_1(t)*, *y\_2(t)* и *y\_ос(t)*.

В том случае, когда математическая модель динамики блока описывается большим числом дифференциальных уравнений, например, 5-ю, то 2-я исполняемая строка будет иметь вид:

**init** x1=0; **init** x2=1; **init** x3=2; **init** z=0; **init** R=0;

где *х1*, *х2*, *х3*, *z*, *R* – динамические переменные, для которых ниже по тексту будут, и должны быть, записаны дифференциальные уравнения (точнее система уравнений) в форме Коши.

Непосредственно дифференциальные уравнения динамики Объекта Управления записаны с 10-ой по 12-ую исполняемые строки, где символ апострофа обозначает производную по времени, а значения коэффициентов a1, a2, a3, b1, b2 передаются в блок через механизм глобальных переменных (см. предыдущий подраздел).

Если блок **Язык программирования** имеет выходы (выходные порты), то одна из исполняемых строк обязательно должна содержать оператор **output**, описывающий выходные сигналы блока, включая имена выходов и их размерности.

В данном примере последняя строка (**output** y\_1;) описывает один выходной сигнал **y\_1** без указания в прямоугольных скобках размерности выходного сигнала.

Если бы блок имел 2 векторных выхода (2-х жильный и 3-х жильный), то последняя исполняемая строка имела бы, например, вид:

**output** z1[2], z2[3];

В этом случае выше по тексту скрипта необходимо было бы определить (рассчитать) все составляющие выходных сигналов, например:

z1[1]=a1+**sin**(y1); z1[2]=**exp**(y2);

z2[1]=y1\*y2; z2[2]=**sqrt**(**abs**(y\_oc)); z2[3]=(y2)^a2;

После ввода всего текста скрипта в окно Редактора переместите курсор на кнопку **Применить** (1-я справа) и сделайте щелчок левой клавишей "мыши": окно Редактора закроется и откроется субмодельное схемное окно, в котором блок **Язык программирования** будет иметь один входной и один выходной порты.

Если бы блок **Язык программирования** формировал два выхода (например, z1 и z2 ⟹ см. выше), то изображение блока на структурной схеме имело бы два выходных порта. При ориентации блока "слева-направо" 1-ый выходной порт будет верхним, а 2-ой выходной порт – нижним.

Завершите оформление схемного окна субмодели, соединив все блоки линиями связи, а также введя в диалоговом окне блока с подписью W₁(s) новое значение коэффициента усиления: **К1**. Структурная схема примет вид, аналогичный рис. 1.7.

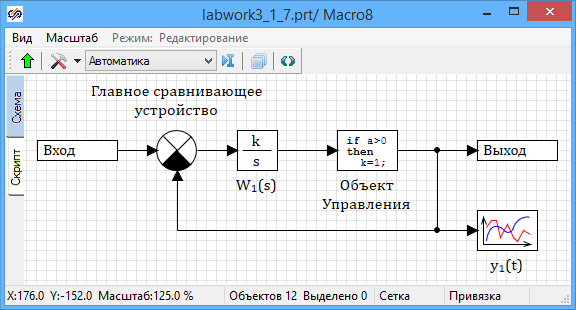


Рис. 1.7

Закройте субмодельное схемное окно (2-х кратным щелчком левой клавиши "мыши" в свободном месте схемного окна) и убедитесь, что в Главном Схемном Окне коэффициент усиление в блоке W₁(s) равен "оптимальному" (k₁ = 0.35). Щелчком "мыши" по кнопке **Пуск** запустите задачу на счет и убедитесь, что результаты расчета для "основной" и "параллельной" САР совпадают абсолютно. Для этого в диалоговом окне **Настройка** блока *График y(t)* установите для 1-ой линии следующие параметры: тип линии – сплошная толщиной 4, а цвет линии – красный, а для 2-ой линии: тип линии – штриховая, цвет – жёлтый, толщина 2; тогда наложенные графики будут иметь вид, близкий к рис. 1.8.



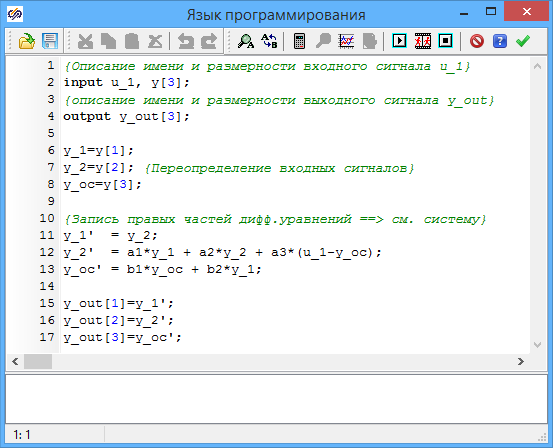
Рис. 1.8

Для создания математической модели рабочего процесса, описываемого сложными нелинейными дифференциальными уравнениями с переменными коэффициентами (например, процессы теплогидравлики в элементах теплофизического оборудования), целесообразнее использовать блок **Язык программирования**, причем иногда – только для формирования правых частей системы дифференциальных уравнений (записанных в форме Коши).

В таком варианте процедуру интегрирования можно реализовать с использованием типового блока **Интегратор** (блок векторизован), расположенного за блоком **Язык программирования**, причем выходной сигнал блока **Интегратор** (обычно векторный) подается на вход блока **Язык программирования**, не образуя алгебраической "петли" (контура).

Реализуем такой подход для формирования математической модели блока Объект Управления в "параллельной" САР.

В верхней части рис. 1.9 представлена экранная копия окна Редактора, в котором реализован алгоритм вычисления правых частей дифференциальных уравнений, описывающих динамику блока Объект Управления.



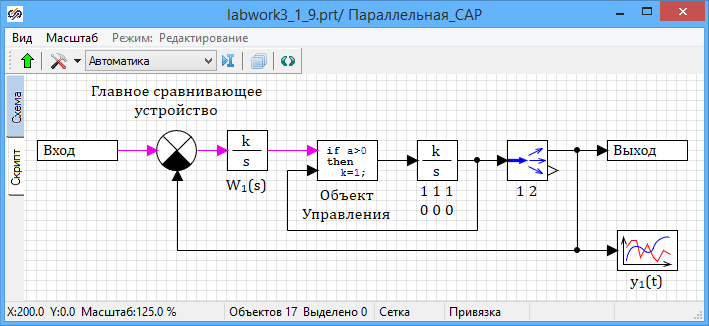


Рис.1.9

Только внешне основной текст (по сравнению с рис. 1.6) изменился незначительно. Кардинальным образом изменился "смысл" символа апостроф: теперь запись y\_1'=... обозначает не первое дифференциальное уравнение, а просто определяет новую переменную с именем **y\_1'**. Появился новый входной сигнал **y**, который является "трёхжильным" (векторным) и вводит в блок значения динамических переменных **y\_1**, **y\_2** и **y\_oc** после каждого шага интегрирования (в том числе и пробного шага). На выходе блока **Язык программирования** сформирован векторный сигнал **y\_out** с размерностью, равной трем ⟹ y = [y1, y2, y\_oc].

Экранная копия субмодельного схемного окна с измененной структурной схемой "параллельной" САР представлена в нижней части рис. 1.9. Подпись типового блока **Интегратор** (в две строки) "сообщает", что данный блок выполняет операцию интегрирования в "векторном" варианте, причем в его диалоговом окне установлены коэффициенты усиления **[1 1 1]** (три числа 1 через пробел или лучше через запятую в квадратных скобках) и нулевые начальные условия **[0 0 0]** (три числа 0 через пробел). Блок **Демультиплексор** "извлекает" из вектора сигнал **y\_1**.

Измените текст в окне Редактора блока «Язык программирования» и скорректируйте структурную схему в субмодельном окне, как это выполнено на рис. 1.9.

Выполните моделирование (щелчок по кнопке **Пуск**) и убедитесь, что графики переходных процессов (формируемые блоком *График y(t)* в Главном Схемном Окне) в "основной" САР и в новой "параллельной" САР совпадают абсолютно.

Сохраните данный вариант проекта (задачи) на жестком диске под новым именем.

## 1.5 Формирование уравнений динамики САР в переменных состояния

Выполним 4-й этап настоящего задания, основной целью которого (этапа) является освоение методов формирования уравнений динамики линейных САР с использованием типового блока **Переменные состояния**. Для этого создадим вторую "параллельную" САР (то есть третью по счету), расположенную в субмодели "Параллельная" САР.

Откройте проект, структурная схема в субмодельном окне которого имеет вид, подобный схеме в нижней части рис. 1.9.

Инициализируйте в "Линейке" типовых блоков закладку **Динамические** звенья и перенесите в субмодельное окно типовой блок **Переменные состояния**. Соединив блок линиями связи и выполнив оформление поясняющих подписей, придайте структурной схеме субмодели вид, аналогичный рис. 1.10.

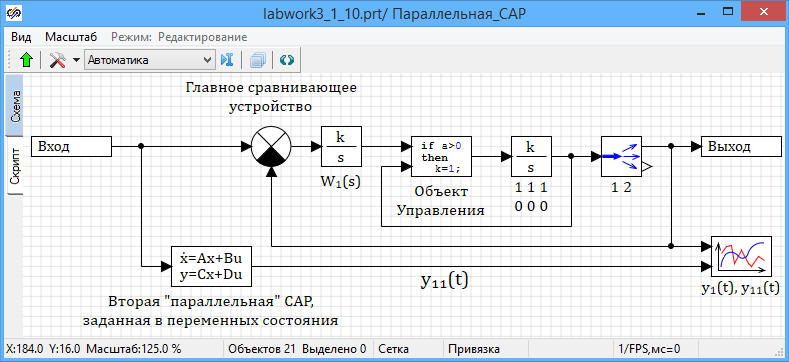


Рис. 1.10

В блоке **Переменные состояния** матрицы А, В, С и D задаются не по строкам (как обычно), а по столбцам.

Матрица входа В, фактически являющаяся вектором-столбцом и нулевая матрица обхода D (см. соотношения (1.5)) не требуют дополнительных пояснений для ввода в соответствующие строки диалогового окна блока Переменные состояния.

Матрица А, определенная для Второй "параллельной" САР соотношениями (1.5), при ее задании в диалоговом окне блока **Переменные состояния** должна представляться как "особая" вектор-строка, содержащая 4 элемента, каждый из которых является числовым вектором и содержит по 4 элемента (числовые или символьные значения соответствующего столбца матрицы А) ⟹ см. ниже.

Матрица С также должна представляться как "особая" вектор-строка, содержащая 4 элемента, каждый из которых является числовым вектором и только в данной задаче содержит по одному элементу ⟹ см. ниже.

Задание свойств блока **Переменные состояния** выполним через механизм Глобальных констант. Переместите курсор на вкладку **Скрипт** (субмодели) и выполните щелчок левой клавишей "мыши". Введите текст, как это выполнено на рис. 1.11.

Поскольку коэффициенты **а1**, **а2**, **а3**, **а4** и **К1** определены как глобальные в окне Редактора Скрипта Главного Схемного Окна, в данном окне Редактора... они используются для задания элементов матрицы А, не равных нулю или единице. Левая часть выражения **const** А1[4] = [0, a21, a31, a41]; определяет, что векторная константа А1 имеет 4 элемента, а правая часть (в прямоугольных скобках) задает значения этих элементов в числовом (**0**) или символьном виде (**а21**).

**Внимание!**

При задании значений элементов любой векторной переменной в символьном виде не допускается знак минус перед символьным элементом.

Например, запись А1[4] = [0 a21 a31 -K1]; является некорректной из-за последнего элемента **-К1**, а именно: при интерпретации выражения произойдёт вычитание a31-K1 и вектор в правой части будет считан интерпретатором как вектор из 3-х элементов. Следует использовать разделители – символы запятой!

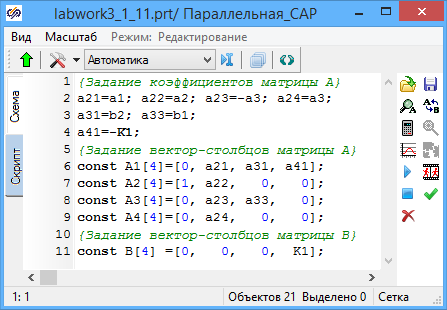


Рис. 1.11

Заполнив окно редактора **Скрипта** субмодели, закройте его щелчком левой клавиши "мыши" по кнопке **Применить**: на экране монитора снова появится субмодельное схемное окно.

Откройте диалоговое окно блока **Переменные состояния** и задайте его Свойства так же, как показано на рис. 1.12, т.е. задавая все элементы матриц А, В, С и D по столбцам. В диалоговых строках любой вектор-столбец заключен в квадратные скобки.



Рис. 1.12

Матрицы А, В, С и D в диалоговом окне можно задавать в более компактном виде, используя векторные переменные (А\_1, А\_2, А\_3, А\_4 и В\_1), определенные как глобальные в окне редактора **Скрипта** ⟹ см. рис. 1.13.

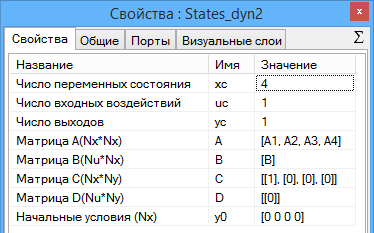


Рис. 1.13

Заполнив диалоговое окно блока **Переменные состояния**, как это выполнено на рис. 1.12, закройте его щелчком левой клавиши "мыши" по кнопке **Ок**.

Запустите задачу на счет и убедитесь, что графики переходных процессов в "параллельной" САР (заданной с использованием блока **Язык программирования**) и во второй "параллельной" САР (заданной с использованием блока **Переменные состояния**), совпадают абсолютно. Для этого в диалоговом окне настройки графика установите для 1-ой линии следующие параметры: тип линии – сплошная двойной толщины, а цвет линии – розовый, а для 2-ой линии: тип линии – пунктирная, цвет – синий. Наложенные графики расчета будут иметь вид, идентичный рис. 1.8.

**Примечание**: графики переходных процессов формируются блоком **Временной график** (подпись y₁(t), y₁₁(t)), расположенным в субмодельном схемном окне (см. рис. 1.10). Абсолютное совпадение графиков можно проверить, переведя график в таблицу (пункт **Таблица** в контекстном меню Графика).

Снова откройте диалоговое окно блока **Переменные состояния** и измените форму задания матриц А и В на компактную (см. рис. 1.13). Повторите процесс моделирования и убедитесь, что графики переходных процессов в "параллельной" САР и во второй "параллельной" САР (с компактным заданием матриц А и В) совпадают абсолютно.

Сохраните данный вариант проекта (задачи) на жестком диске под новым именем.

## 1.6 Реализация "беспроводной" передачи данных

Типовой блок **В память**: основное назначение этого блока – реализация 1-го этапа "беспроводной" передачи данных из одной части структурной схемы в другую. Соответственно, типовой блок **Из памяти** реализует 2-ой этап "беспроводной" передачи данных. Если применить радиотехническую аналогию, то в механизме "беспроводной" передачи данных блок **В память** является "передатчиком", а блок **Из памяти** – "приемником".

На самом деле в среде SimInTech реализован следующей механизм. После переноса в схемное окно блока **В память** и присвоения имени переменной в оперативной памяти ПЭВМ резервируется некий динамический массив. При запуске задачи на счет по размерности сигнала на входе блока **В память** определяется размерность именованной переменной и под нее выделяется соответствующий объем оперативной памяти, в который в процессе моделирования будут записываться данные на каждом шаге расчета (шаге интегрирования).

Если в структурной схеме (например, в каком-то субмодельном окне) присутствует типовой блок **Из памяти**, то с его помощью можно считать (от слова читать) именованные данные из оперативной памяти и выдать их на выход блока **Из памяти**. Затем посредством линий связи данные могут быть переданы на вход другого блока, расположенного в этом схемном окне.

Освоение процедур организации "беспроводной" передачи данных выполним в рамках следующего небольшого задания.

Убедитесь, что проект, в котором структурная схема в субмодельном окне имеет вид, аналогичный рис. 1.10, не закрыт и перейдите в Главное Схемное Окно, выполнив 2-х кратный щелчок левой клавиши "мыши" в свободном месте субмодельного окна. Если Вы успели закрыть проект, то откройте его...

Переместите курсор на линию связи, передающей сигнал управляющего воздействия на вход субмодели "Параллельная" САР и выполните щелчок правой клавишей "мыши" и затем в контекстном меню выберите опцию Удалить линию в связи: ответвление линии связи к субмодели "Параллельная" САР будет удалено.

Аналогичным образом удалите связь (первую и единственную линию связи), соединяющую субмодель "Параллельная" САР и блок с подписью График y(t).

Инициализируйте библиотеку **Субструктуры** и перенесите в Главное Схемное Окно **Субмодель**, разместив её, например, в правом нижнем угле схемного окна. Введите подпись "Копия" для этой субмодели (см. рис. 1.14).



Рис. 1.14

Удалите субмодель "Параллельная" САР (щелчок "мышью" по блоку и затем кнопка Вырезать).

Откройте схемное окно субмодели "Копия", далее переместите курсор на кнопку Вставить (4-я слева в Дополнительной панели инструментов субмодельного схемного окна) и затем выполните щелчок левой клавишей "мыши" в поле субмодельного окна: ранее вырезанная субмодель "Параллельная" САР появится в субмодельном окне 1-го уровня вложенности, а "внутреннее содержание" субмодели "Параллельная" САР (см. рис. 1.10) перейдет во 2-ой уровень вложенности. Проверьте это, открыв субмодель "Параллельная" САР.

Перенесите в субмодельное схемное окно "Копия" из библиотеки Субструктуры блок **В память**, расположив его справа от субмодели "Параллельная" САР. Откройте его диалоговое окно и измените имя переменной на Выход\_Y.

Перенесите в субмодельное схемное окно "Копия" блок **Из памяти**, расположив его слева от субмодели "Параллельная" САР: блок пока не имеет ни пиктограммы, ни текста внутри него – только зеленый фон... Откройте его диалоговое окно (2-х кратный щелчок "мышью"). Выделите в окне Список-источник переменную Вход (см. рис. 1.15) и затем щелкните по красной одинарной стрелки (вниз): переменная Вход перейдет в окно Список-приемник (см. рис. 1.16).

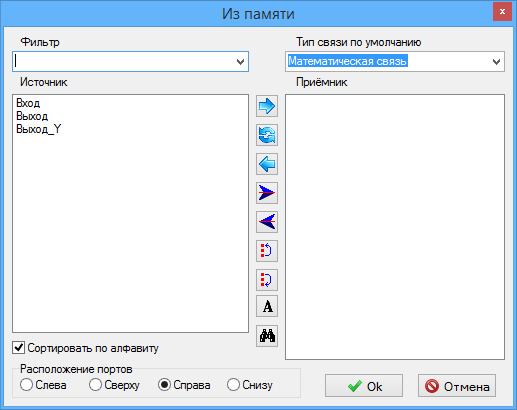


Рис. 1.15



Рис. 1.16

Поясним назначение кнопок (в виде стрелок) в диалоговом окне типового блока **Из памяти**:

- голубая одинарная стрелка налево (см. рис. 1.16) реализует перемещение выбранной переменной из перечня Приёмник в перечень Источник;

- синяя большая стрелка вправо реализует перемещение всех переменных, находящихся в окне Список-источник в окно Список-приемник, а синяя большая стрелка налево – перемещение всех переменных, находящихся в окне Список-приемник в окно Список-источник;

- две крайние кнопки снизу предназначены для принудительной сортировки переменных в окне Список-приемник (если переменных ≥ 2).

Внимание!

1. При закрытии диалогового окна блока **Из памяти** количество выходных портов на блоке будет равно количеству переменных в окне Список-приемник.

2. Блоки **В память** и **Из памяти** реализуют "беспроводную" передачу как скалярных, так и векторных данных.

Соедините линиями связи блоки **Из памяти** и **В память** с субмоделью "Параллельная" САР, как это выполнено на рис. 1.17.



Рис. 1.17

Вернитесь в Главное Схемное Окно и перенесите в него блок **Из памяти**, расположив его приблизительно на то место, где ранее была размещена субмодель "Параллельная" САР. Откройте диалоговое окно блока **Из памяти** и переместите в окно Список-приемник переменную Выход\_Y. Соедините выход блока **Из памяти** с блоком **График y(t)**. Структурная схема примет вид, аналогичный рис. 1.18.



Рис. 1.18

Отметим основные этапы выполненных преобразований структурной схемы:

Была создана новая субмодель с условным названием "Копия".

Субмодель "Параллельная" САР была "отрезана" от структурной схемы "основной" САР и перенесена в субмодель "Копия".

В субмодельное окно "Копия" из библиотеки Субструктуры были перенесены 2 новых блока и соединены линиями связи с субмоделью "Параллельная" САР, причем блок **В память** предназначен для "беспроводной" передачи в Главное Схемное Окно данных о поведении переменной y₁(t) (см. рис. 1.10), а блок **Из памяти** – для "беспроводного" приема из Главного Схемного Окна данных о поведении управляющего воздействия u(t).

В Главное Схемное Окно из библиотеки Субструктуры был перенесен новый блок **Из памяти** и соединен линией связи с блоком График y(t). Блок **Из памяти** реализует "беспроводный" прием из субмодельного окна "Копия" данных о поведении переменной y₁(t).

В результате изменений структурной схемы САР, в целом, субмодель "Параллельная" САР принимает информацию об управляющем воздействии u(t) и передает информацию о поведении переменной y₁(t) не посредством традиционных линий связи, а используя механизм "беспроводной" передачи данных.

Запустите задачу на счет и убедитесь, что графики переменных y(t) и y₁(t) совпадают абсолютно.

На этом "знакомство" с процедурами реализации "беспроводной" передачи данных завершено и можно переходить к выполнению самостоятельной части лабораторной работы.

# 2 САМОСТОЯТЕЛЬНОЕ ИССЛЕДОВАНИЕ ДИНАМИКИ САР ЯР, ЗАДАННОЙ В ПЕРЕМЕННЫХ СОСТОЯНИЯ И В ФОРМЕ КОШИ

В процессе выполнения лабораторной работы №1 Вы сформировали структурную схему простейшей математической модели динамики САР ядерного реактора. Поскольку задача (проект) была Вами сохранена на жестком диске, откройте "свою" модель динамики САР ЯР, внешний вид структурной схемы которой (с точностью до Ваших художественно-оформительских способностей) имел вид, приблизительно соответствующий структурной схеме на рис. 2.1.



Рис.2.1

Подписи блоков, которые формируют преобразование и отображение сигналов (см. рис. 2.1) дают минимальную информацию, по которой Вы должны "вспомнить" цель задания в лабораторной работе №1 и примененные Вами методы ее решения ...

Тем не менее, необходимо повторить исходные уравнения и соотношения, использование которых позволило Вам выполнить лабораторную работу №1 и которые, несомненно, необходимы Вам для выполнения самостоятельного исследования динамики САР ЯР в настоящей лабораторной работе.

Переход к нормированным отклонениям переменных и последующая линеаризация дифференциального уравнения для нормированных отклонений плотности нейтронов, позволяют представить математическую модель точечной кинетики нейтронов с одной эффективной группой запаздывающих нейтронов в следующем виде:

|  |  |
| --- | --- |
|  | (2.1) |

где: – эффективная доля запаздывающих нейтронов;

– время жизни мгновенных нейтронов;

– постоянная распада ядер-предшественников запаздывающих нейтронов;

– нормированные отклонения концентрации ядер-предшественников запаздывающих нейтронов.

Местная обратная связь, определяемая отрицательным температурным эффектом реактивности, описывается следующими уравнениями:

|  |  |
| --- | --- |
|  | (2.2) |

где: – температурный коэффициент реактивности;

– стационарная температура топлива в активной зоне и нормированное отклонение температуры топлива в активной зоне от стационара, соответственно;

– постоянная времени (инерционность) топлива в активной зоне;

– безразмерный коэффициент.

**Привод регулирующего стержня** (см. рис. 2.1) состоит из электродвигателя постоянного тока, редуктора, муфт, преобразователя движения, непосредственно регулирующего стержня и т.п., однако для упрощения задачи все эти элементы объединены в одно звено.

Нестационарные процессы в блоке (в звене) **Привод регулирующего стержня** описываются следующим дифференциальным уравнением:

|  |  |
| --- | --- |
|  | (2.3) |

где – коэффициент скоростной эффективности; – постоянная времени (инерционность) **Привода регулирующего стержня.**

Необходимо заметить, что хотя система уравнений (2.1) и приведена к стандартной форме Коши, для включения ее в полную систему уравнений (описывающих динамику всех элементов рассматриваемой САР ЯР) необходимо выразить изменение реактивности через ее составляющие .

Уравнения, описывающие динамику местной обратной связи, требуют "косметической" редакции, а уравнение динамики **Привода регулирующего стержня** – перехода от описания в переменных "вход-выход" к переменным состояния.

В самостоятельной части настоящей лабораторной работы каждой подгруппе необходимо выполнить следующие этапы:

1. Выполнить необходимые преобразования и записать математическую модель динамики линеаризованной САР в переменных состояния, вычислив все элементы соответствующих матриц (А, В, С и D) и векторов (каких?!) в символьном виде.
2. Преобразовать структурную схему САР (см. рис. 2.1), добавив к ней "параллельную" САР ядерного реактора, описываемую полностью в переменных состояния (с использованием типового блока **Переменные состояния** и заданием матриц А, В, С и D через механизм Глобальных констант/переменных в компактной форме).
3. Выполнить моделирование переходных процессов в "основной" САР (описываемой в переменных "вход-выход") и в "параллельной" САР (описываемой в переменных состояния) при подаче управляющего воздействия , построив на одном Графике временные зависимости для обеих САР, а на другом Графике временные зависимости для обеих САР, соответственно (используя наложение графиков).
4. Выполнить моделирование переходных процессов в "основной" САР (описываемой в переменных "вход-выход") и в "параллельной" САР (описываемой в переменных состояния) при подаче возмущения по реактивности , построив на одном Графике временные зависимости для обеих САР, а на другом Графике временные зависимости для обеих САР, соответственно (используя наложение графиков).
5. Сформировать в среде SimInTech новый проект (задачу), описывающий математическую модель динамики линеаризованной САР в переменных состояния с использованием типового блока Переменные состояния, а математическую модель нелинеаризованной САР – в форме Коши с использованием блока **Язык программирования**.

**Примечания:**

* + Формирование в новом проекте математической модели динамики линеаризованной САР в переменных состояния целесообразно реализовать, используя процедуры копирования в следующей последовательности: сначала скопировать в окно **редактора Скрипта Проекта** (Субмодели) нового проекта содержание аналогичного окна из предыдущего проекта, а затем скопировать в Схемное окно нового проекта блок **Переменные состояния** из предыдущего проекта.
  + Математическая модель нелинеаризованной САР получается из уравнений динамики линеаризованной САР. Для этого необходимо "вернуть" в линеаризованные уравнения составляющие, отброшенные при линеаризации, как составляющие 2-го порядка малости (см. лекции по курсу "Управление в технических системах").

1. Выполнить моделирование переходного процесса в линеаризованной САР и в нелинеаризованной САР при подаче управляющего воздействия , сравнив на одном графике временные зависимости , а на другом графике временные зависимости для обоих вариантов математических моделей динамики САР, соответственно (используя наложение графиков).
2. Выполнить моделирование переходного процесса в линеаризованной САР и в нелинеаризованной САР при подаче возмущающего воздействия , сравнив на одном графике временные зависимости , а на другом графике временные зависимости для обоих вариантов математических моделей динамики САР, соответственно (используя наложение графиков).