Приложение 2

Руководство по работе с пакетом динамического моделирования VisSim 3.0

**Руководство по работе с пакетом динамического моделиgрования VisSim 3.0**

1.1. Запуск и описание основного окна программы VisSim.

Запуск программы VisSim можно произвести через меню «Пуск» («Start») в Windows. Для этого следует щелкнуть левой кнопкой мыши на этой кнопке, затем выбрать пункт главного меню «Программы». Если на компьютере установлена, например, версия 3.0 (FAP) пакета VisSim, то, как правило, чтобы запустить основную программу, необходимо последовательно выбирать следующие пункты меню «Пуск» (начиная с самой кнопки «Пуск»):

Пуск -> Программы -> VisSim FAP - VisSim 3.0 -> VisSim 3.0 (FAP)

Щелчок по ярлыку VisSim 3.0 запускает программу на исполнение.

***Примечание****: В дальнейшем под щелчком мыши подразумевается щелчок именно левой кнопкой мыши. Если требуется выбрать какой-либо пункт меню, то необходимо навести курсор мыши на этот пункт и щелкнуть левой кнопкой.*

 После загрузки на экране появляется окно программы, которое включает в себя основные элементы (начиная с самого верхнего, см. рис. 1.1.):

- ***заголовок окна***, в котором указано название программы «VisSim» и название текущей диаграммы. Первоначально никакой диаграммы в VisSim не загружено, поэтому можно видеть чистый лист, название которого по умолчанию - «Diagram1».

- ***строка меню***, состоящего из пунктов «File», «Edit», «Simulate», «Blocks», «Comm», «Analyze», «View» и «Help» (описание пунктов меню приведено в п.2).

- ***панель инструментов***. На это панели расположены кнопки, каждая из которых соответствует какому-либо действию. Например, первая кнопка создает новый пустой лист диаграммы, вторая - позволяет открыть уже имеющуюся на диске диаграмму и т.д. Все эти кнопки дублируют некоторые из команд основного меню и предназначены для обеспечения большего удобства работы с пакетом.

На панели инструментов также располагаются некоторые палитры типовых блоков.

- ***дерево диаграммы***, показывающее иерархическую структуру блоков диаграммы.

- ***окно диаграммы***. В этом окне находится текущий лист диаграммы (первоначально он пуст). Линии прокрутки справа и внизу окна диаграммы позволяют перемещать видимую область листа для просмотра больших диаграмм.



Рисунок П 2.1$-$ Окно основной программы VisSim 3.0.

- ***строка состояния***. В этой, самой нижней строке экрана, отображается информация о текущем состоянии системы:

Blks - указывает число функциональных блоков на диаграмме,

Rng - начальный и конечный моменты времени расчета,

Step - шаг по времени,

Т(Time) - текущее время расчета (если диаграмма не запущена на выполнение, то время равно нулю).

 Также в строке указывается метод интегрирования. Пакет VisSim позволяет просчитывать дифференциальные уравнения следующими методами: Euler - метод Эйлера, Trapezoidal - метод трапеций, RK2 - Рунге-Кутта 2-го порядка, RK4 - Рунге-Кутта 4-го порядка, Adaptive RK5 -адаптивный метод Рунге-Кутта 5-го порядка, Adaptive Bulirsh-Stoer и Backward Euler.

**Замечание**. Желательно, чтобы не иметь в дальнейшем проблем со шрифтами, сразу после загрузки программы изменить их кодировку. Для этого следует выбрать пункт «View -> Fonts…». Появившееся окно определяет шрифт, который используется по умолчанию (см. рис. 1.2.). Здесь можно выбрать тип шрифта, его начертание, размер, цвет и кодировку. По умолчанию установлен западноевропейский набор символов. Если на вновь созданной диаграмме будут расположены надписи на русском языке, то необходимо выбрать в качестве набора символов кириллицу.



Рисунок П 2.2$-$ Выбор шрифта.

 **1.2. Принципы построения и редактирования диаграмм.**

Под диаграммой понимается набор функциональных блоков, соединенных между собой линиями связи и отображаемый в окне диаграммы VisSim.

Палитры функциональных блоков находятся в меню «Blocks» и представляют собой достаточно большой набор основных элементов автоматики. Каждый из блоков моделирует динамические свойства некоторого соответствующего ему простого объекта. Размещая эти объекты на диаграмме и соединяя их функциональными связями, можно моделировать более сложные системы.

Любая диаграмма может быть либо загружена с диска (если она была ранее создана и сохранена в виде файла), либо создана заново. Процесс создания диаграммы заключается в выборе мышью соответствующих блоков из палитр и размещении их на листе диаграммы с последующим их соединением связями.

Для **размещения** какого-либо блока на диаграмме необходимо выполнить следующие действия мышью: щелкнуть на меню «Blocks», выбрать соответствующую палитру, выбрать соответствующий блок и щелкнуть на нем мышью. Если блок продублирован в виде соответствующей кнопки на панели инструментов, достаточно щелкнуть по ней мышью.

После этого остается щелкнуть на том месте диаграммы, куда необходимо поместить блок. Сразу же блок появится на экране.

Например, расположение на диаграмме усилительного блока (блок «gain») происходит в следующей последовательности: щелчок на пункте меню «Blocks» -> выбор палитры «Arithmetic» -> выбор пункта «gain» -> щелчок на пункте «gain» -> подведение курсора к тому месту диаграммы, где будет располагаться блок -> щелчок мыши размещает блок на диаграмме.

Все операции производятся левой кнопкой мыши. Правая кнопка в VisSim предназначена только для изменения свойств отдельных блоков. Если щелкнуть на каком-либо блоке правой кнопкой, то появится окно с его настройками. Например, окно настроек усилительного блока (блок «gain») содержит только одну настройку - коэффициент усиления. Вновь созданный блок содержит только настройки, принятые по умолчанию. Например коэффициент усиления усилительного блока по умолчанию равен 1.

Каждый блок изображается на диаграмме, как правило, в виде прямоугольника, внутри которого схематично изображено его функциональное назначение. С левой стороны блока находятся его входные сигналы (изображены в виде стрелок), с правой выходные. Количество входов и выходов у каждого блока различно и, как правило, неизменно.

Блоки можно **перемещать** на диаграмме для их наиболее удобного расположения.

***Примечание****: Для этого необходимо нажать (не щелкнуть!) левую кнопку мыши на блоке, который следует переместить. Далее, не отрывая пальца от кнопки, переместить мышь в нужную точку. После отпускания кнопки мыши блок расположится в том месте, где был курсор.*

Для одновременного перемещения группы блоков необходимо начала эту группу выделить.

**Выделение** блоков производится следующим образом. Мысленно нужно представить себе прямоугольник на экране, который бы окружал (или хотя бы задевал) только те блоки, которые следует выделить. Далее необходимо нажать левую кнопку мыши в левом верхнем углу этого прямоугольника и, не отпуская ее, начертить этот прямоугольник, проведя мышью с нажатой кнопкой в противоположный, правый нижний, угол. (Прямоугольник, как правило, отображается зеленым цветом). Как только кнопка мыши будет отпущена, блоки окажутся выделенными. (Блоки почернеют или инвертируют цвет). Далее с группой блоков можно оперировать как с единым блоком. Например, ее можно перемещать, удалять, копировать и объединять.

Для **удаления** блоков следует сначала выделить, как описано выше, блок (или блоки) и нажать на клавишу «Delete» клавиатуры.

Если на строящейся диаграмме должны иметь место несколько групп блоков, похожих по структуре и (или) составу (повторяющиеся элементы схемы), то значительно упростить процесс ее создания поможет **копирование** блоков. Для копирования необходимо выделить копируемую группу блоков и нажать комбинацию клавиш «Ctrl+C» (то есть, нажать клавишу «Ctrl» и, удерживая ее в нажатом положении, нажать «С»). Группа блоков скопируется в буфер Windows. Далее нажимается комбинация «Ctrl+V» и мышью указывается место, куда будет помещена скопированная группа. Щелчок мыши располагает группу на диаграмме. Если необходимо разместить третью, четвертую и т.д. группы, то повторные нажатия «Ctrl+V» и щелчки мыши позволят это сделать.

**Объединение** блоков применяется для визуального упрощения диаграммы. Например, если диаграмму можно разбить на несколько структурно отдельных объектов (например, модель технологического объекта, модель регулятора и т.д.) и при этом структура диаграммы достаточно сложна, то каждую группу блоков можно объединить в единый блок. Для этого необходимо выделить группу блоков, войти в меню «Edit» и щелкнуть на пункте «Create Compound Block». В появившемся окне следует ввести название создаваемого объекта и нажать кнопку «ОК». Сразу же группа блоков на диаграмме будет заменена одним блоком зеленого цвета. Если войти в этот блок (щелчком правой кнопки мыши или двойным щелчком левой кнопки), то можно увидеть первоначально располагавшиеся на диаграмме блоки. Количество вложений блоков не ограничено.

**Создание линий связи** производится левой кнопкой мыши. Чтобы соединить два расположенных на диаграмме блока необходимо подвести мышь к выходу одного блока. Как только курсор превратиться в стрелку вида ↑, нажать левую кнопку. Далее, удерживая кнопку нажатой, подвести мышь ко входу второго блока. В процессе движения мыши за ней будет тянуться линия связи. Как только курсор достигнет входа, отпустить кнопку. Практика показывает, что линии удобнее проводить в обратном порядке

**Для разрыва линий** **связи** необходимо навести мышь на начало (или конец) линии, нажать левую кнопку, немного отодвинуть мышь от блока и отпустить кнопку. Линия должна исчезнуть.

Иногда (например, при изображении линий обратной связи) требуется **развернуть блок** (блоки), то есть расположить его так, чтобы его входы находились с правой стороны, а выходы - с левой. Для этого необходимо выделить блок (блоки) и нажать комбинацию «Ctrl+←» или «Ctrl+→» или выбрать в пункт «Flip Horizontal» (в версии 2.0 - «Rotate 180») в меню «Edit».

 **1.3. Сохранение диаграммы на диске.**

Созданную или отредактированную диаграмму можно сохранить на диске. по команде меню «File -> Save». Этому пункту соответствует кнопка «Save» на панели инструментов (см. рис. 1.3., 3-я кнопка). Щелчок мышью по этой кнопке будет иметь тот же эффект.

Если требуется сохранить диаграмму под каким-либо новым именем, то нужно выбрать пункт «Save as…» того же меню «File». В открывшемся окне нужно набрать имя диаграммы. Можно также указать место сохранения диаграммы на диске. После того, как все сделано, нажать кнопку «ОК». Диаграмма сохранится на диске в виде файла с расширением .vsm.


Рисунок П2.3 $-$ Панель инструментов.

 **1.4. Открытие диаграммы.**

Чтобы открыть диаграмму, ранее сохраненную на диске, необходимо выбрать пункт меню «File -> Open…» или нажать на кнопку «Open» панели инструментов (см. рис. 1.3., 2‑я кнопка). При этом появится окно, отображающее содержимое диска, в котором нужно найти файл, выделить его щелчком мыши и нажать кнопку «ОК».

 После этого VisSim выдаст окно, в котором спрашивается, сохранять ли изменения сделанные в уже открытой диаграмме. Если никаких важных изменений не было сделано, то нужно нажать кнопку «Cancel» («Отмена»), в противном случае - «ОК». Сразу после этого имеющаяся диаграмма на экране будет заменена на новую.

 **1.5. Создание новой диаграммы.**

Для создания новой диаграммы выбирается пункт меню «File -> New» или нажимается кнопка «New» (см. рис. 1.3., 1-я кнопка). После щелчка мыши VisSim выдаст окно, спрашивающее о том, сохранять или не сохранять сделанные изменения (см. п. 1.4.).

 **1.6. Печать диаграммы.**

 **1.6.1 Настройка печати.**

Перед печатью необходимо настроить параметры страницы, то есть задать ее размер, поля и расположение (горизонтальное или вертикальное). Для этого выбирается пункт меню «File -> Page setup…». В открывшемся меню указываются следующие параметры:

- размеры полей: Right - правое поле, Left - левое, Top - верхнее, Bottom - нижнее. Для изменения значений полей следует щелкнуть на соответствующем белом поле и ввести значение.

- расположение страницы (поле Orientation): Portrait - вертикальное, Landscape - горизонтальное.

- заголовок страницы (поле Header) и нижний колонтитул (Footer).

- размер (Size) и источник бумаги (Source).

- если отметить галочкой пункт «Fit to page» (щелчком в квадратном поле слева от пункта), то при печати масштаб диаграммы будет подобран под размеры листа.

После того как установлены все настройки необходимо нажать «ОК».

Нажатие на кнопку «Print…» позволяет выбрать принтер, на котором будет печататься диаграмма.



Рисунок П2.4$ -$ Окно параметров печати.



Рисунок П2.5$ -$ Окно печати.

 **1.6.2 Предварительный просмотр.**

Если все настройки печати установлены правильно, то перед печатью диаграммы желательно просмотреть, как она будет выглядеть на бумаге. Для этого используется пункт «File -> Print Preview…». Сразу после щелчка на нем появляется изображение страницы.

 **1.6.3 Вывод на печать.**

Для окончательной печати текущей диаграммы выбирается пункт меню «File -> Print…». В появившемся окне можно указать настройки, которые будут действовать только в данном сеансе печати:

- в настройке «Print Quality» можно указать качество печати (для современных струйных и лазерных принтеров это не существенно). Для этого нужно щелкнуть на небольшой кнопке с треугольной стрелкой справа от значения настройки и в выпавшем списке щелчком мыши указать нужное качество. Качество указано в dpi (чем больше dpi, тем четче изображение).

- если требуется напечатать несколько копий диаграммы, то число копий указывается в пункте «Copies».

- параметр «Print to File» позволяет выводит диаграмму не на принтер, а в файл для последующей печати, например, на другом компьютере.

- параметр «Fit to Page» позволяет уместить диаграмму в одну страницу.

Если нажать кнопку «Setup», то появляется окно настроек принтера, в котором можно указать размер бумаги, ориентацию, текущий принтер и т.д.

После изменения необходимых настроек следует нажать кнопку «ОК», в результате чего диаграмма будет послана на печать. Если нажать «Cancel», то печать будет отменена.

**2. Основное меню VisSim.**

 **2.1. Пункт File.**



Рисунок П2.6

**New -** создание новой диаграммы,

**Open** - открытие диаграммы,

**Save** - сохранение на диске,

**Save as…** - сохранение на диске по каким-либо новым именем,

**Diagram Information…** - выдает окно с информацией о текущей диаграмме: Заголовок, автор, количество блоков и др.

**Page Setup…** - поля страницы,

**Print…** - печать диаграммы,

**Printer Setup…** - вызывает окно настроек драйвера принтера,

**Printer Preview**… - предварительный просмотр диаграммы перед печатью,

**Real Time Config** - изменение настроек драйвера реального времени,

**Exit** - выход из программы.

Далее в меню идет список диаграмм, которые были открыты недавно. Щелчок по каждой из них вызывает их на экран.

 **2.2. Пункт Edit.**


Рисунок П2.7

**Undo** - отмена последнего произведенного действия,

**Cut** - вырезание блока или группы блоков (блоки удаляются с диаграммы и помещаются в буфер),

**Copy** - копирование блоков в буфер (блоки не удаляются с листа),

**Paste** - вставка блоков из буфера,

**Clear** - удаление блоков (в отличие от «Cut» не помещает блоки в буфер),

**Flip Horizontal** - разворот блока на 180°,

**Create Compound Block** - объединение группы блоков в единый блок,

**Dissolve Compound Block** - действие, обратное «Create Compound Block»,

**Find…**  - поиск переменной на диаграмме (в появляющемся окне необходимо ввести имя переменной, т.е. имя блока типа variable; после нажатия на «ОК» соответствующий блок будет выделен),

**Replace… -** поиск с заменой,

**Block Properties…** - изменение настроек какого-либо блока (после выбора этого пункта курсор мыши превращается в крест; если этот крест навести на какой-либо блок и нажать левую кнопку, то появится окно со списком настроек этого блока; команда аналогична нажатию правой кнопки мыши на блоке),

**Add Connector** - добавление еще одного входа (или выхода) к какому-либо блоку (после выбора курсор мыши превращается в крест, которым нужно указать на нужный блок и щелкнуть левой кнопкой; если функция блока позволяет это сделать, то у него появится новый вход или выход),

**Remove Connector** - команда, противоположная «Add Connector» (курсор также превращается в крест, которым нужно указать на блок с удаляемым входом)

**Repaint Screen** - перерисовка экрана (позволяет удалить графический «мусор», иногда появляющийся на диаграмме),

**Preferences…** - настройка внешнего вида окна программы (показывать ли линии прокрутки, привязывать ли блоки к сетке и т.д.),

**Toolbar…** - настройка панели инструментов (вызывается окно, в котором можно настроить несколько дополнительных кнопок панели, что позволяет облегчить и ускорить процесс создания диаграмм).

 **2.3. Пункт Simulate.**



Рисунок П2.8

Пункт содержит команды и настройки, используемые при расчете:

**Go** - запуск модели на исполнение (аналогично нажатию на кнопку с зеленой стрелкой на панели задач),

**Stop** - приостановить процесс расчета,

**Single Step** - выполнить один шаг расчета,

**Continue** - продолжить приостановленный расчет,

**Reset** - сброс результатов,

**Simulation Properties…** - выводит окно с настройками расчета. В окне можно задать следующие параметры:

 На закладке Range*:*

Start - начальное значение времени,

Frequency - частота выполнения шагов расчета,

End - конечное значение времени.

На закладке Integration Method- алгоритм интегрирования дифференциальных уравнений. В этой настройке нужно мышью щелкнуть на нужном алгоритме. Методы интегрирования перечислены в п. 1.1.

На закладке Defaults *-* настройки, принимаемые по умолчанию для вновь создаваемых диаграмм (Range Start, Frequency, End, Integration Method, Max Plotted Points- число выводимых на график точек).

**Optimization Properties…** настройки, используемые в процессе оптимизации (метод, число итераций, погрешность)

 **2.4. Пункт Blocks.**

Пункт содержит палитры функциональных блоков. Чтобы выбрать какой-либо блок, необходимо войти в это меню, найти нужный блок в какой-либо палитре и щелкнуть на нем. После этого можно щелчком указать на его место на диаграмме. Подробнее о палитрах см. п. 3.

 **2.5. Пункт Analyze.**

Пункт содержит команды, используемые при анализе сформированных систем: определение передаточных функций систем, полюсов (корней характеристических полиномов), нулей функций, частотных характеристик (АЧХ и ФЧХ), устойчивости по Найквисту и др. Подробнее см. п. 4.

 **2.6. Пункт View.**



Рисунок П2.9

В этом пункте находятся настройки внешнего вида программы и диаграмм:

**Fonts…** - настройка используемого на диаграмме шрифта (размер, вид, кодировка и цвет).

В поле «Шрифт» можно выбрать вид шрифта щелчком левой кнопки мыши. В поле «Начертание» - тип начертания: обычный, курсив, полужирный, полужирный курсив.

Также можно выбрать размер, цвет и набор символов. Набор символов должен быть установлен как «Кириллица».

**Colors…** - настройка цветовой гаммы внешнего вида программы (цвета надписей, фона и др.),

**Presentation Mode** - режим презентации (более нагляден),

**Display Mode** - если возле него стоит галочка, то на диаграмме все линии связи делаются невидимыми,

**Block Labels** и **Connector Labels** позволяют разрешать или запрещать отображение подписей к блокам и линиям связи.

**Status Bar** - отображает или запрещает отображение строки состояния внизу окна,

**ToolBar** - отображает или запрещает отображение панели инструментов.

 **2.7. Пункт Help.**



Рисунок П2.10

В этом пункте выводится помощь по программе:

**Contents** - содержание,

**Search…** - поиск,

**Wiring** - информация о работе с функциональными связями,

**Simulating -** информация о моделировании динамики,

**Using Help -** использование помощи,

**About VisSim** - информация о версии программы и др.

 **3. Краткое описание функциональных блоков VisSim.**

В данном разделе функциональные блоки упорядочены по палитрам, приведен перечень блоков по алфавиту(подробное описание блоков в п. 5).

**Animation** – палитра блоков анимации:

animate блок анимации

lineDraw

**Annotation** – пояснения с диаграмме

bezel картинка из файла для вставки на лист диаграммы

comment пояснение на диаграмме (несколько строк)

date вставка текущей даты

index

label комментарий (одна строка)

scalarToVec перевод скалярной величины в векторный вид

vecToScalar перевод секторной величины в скалярный вид

variable переменная (применение переменных позволяет уменьшить загроможденность диаграммы линиями связи)

wirePositioner позиционер для формирования направления линий связи

**Arithmetic** – арифметические функции

\* (multiply) умножение двух сигналов (имеет два входа и один выход)

-X (negate) инверсия знака

/ (divide) деление

abs модуль

gain блок усиления сигнала (коэффициент усиления)

pow

sign знак сигнала (+ или -)

summingJunction сумматор

**Boolean** – логические функции

< меньше

<= меньше или равно

== равенство

!= не равно

> больше

>= больше или равно

and И (логическая конъюнкция)

not НЕ (логическое отрицание)

or ИЛИ (логическая дизъюнкция)

xor отрицание ИЛИ (XOR)

**DDE** – блоки для организации обмена между программами

DDE блок обмена данными с другой программой

DDEreceive блок приема данных из другой программы

DDEsend блок отсылки данных в другую программу

**Integration** – блоки интегрирования

integrator интегрирующее звено

limitedIntegrator ограниченный интегратор (имеет максимальное и минимальное значения выходного сигнала)

resetIntegrator сброс интегратора

**Linear System** – линейные системы

stateSpace пространство состояний

transferFunction передаточная функция

**Nonlinear** – нелинейные системы

case оператор варианта (имеет несколько входов плюс вход case, определяющий какой из входных сигналов подавать на выход)

crossDetect реле, определяющее сигналом какого-либо уровня

deadband реле с зоной нечувствительности

int целая часть числа

limit ограничитель сигнала

mapmax определение максимального значения сигнала

min определение минимального значения сигнала

quantize дискретизатор

relay «замораживатель» выходного сигнала (имеет два входа: х и b, при подаче на вход b сигнала на выходе блока устанавливается постоянный сигнал, равный мгновенному значению сигнала х).

**Optimization** – оптимизация систем

constraint

cost стоимость (для анализа)

globalConstraint

parameterUnknown

unknown

**Random Generator** – генераторы случайных сигналов

gaussian генератор случайно сигнала по Гауссу

uniform

**Real Time** – функции реального времени

rt-DataIn

rt-DataOut

**Signal Consumer** – блоки для отображения сигналов

display индикатор цифровой

error

export

histogram гистограмма

light лампочка (один вход)

meter стрелочный индикатор

plot графики (выводит одновременно до 4-х графиков разными цветами)

stop

stripChart полосатая диаграмма

**Signal Producer** – генераторы сигналов

button кнопка (один выход)

const константа

import получение данных из файла

parabola параболическое воздействие

pulseTrain импульсное воздействие

ramp линейное воздействие

realTime настоящее время

sinusoid синусоидальный сигнал

slider ползунок (для генерации входных сигналов)

step шаг

**Time Delay** – запаздывания

timeDelay запаздывание (два входа: задерживаемый сигнал и величина задержки)

**Transcendental** – тригонометрические функции

acos арккосинус

asin арксинус

atan2 арктангенс

bessel бессель

cos косинус

cosh гиперкосинус

exp экспонента

log10 десятичный логарифм

ln натуральный логарифм

sin синус

sinh гиперсинус

sqrt корень

tan тангенс

tanh гипертангенс

**UserFunction** функция, определенная пользователем

**NeuralNet** блок нейронной сети

**4. Примеры простых проектов.**

 **4.1. Построение переходной кривой.**

**Задание.** Требуется построить переходную кривую объекта, заданного передаточной функцией:

.

**Решение.** Для построения переходной кривой создается схема, представленная на рис. 4.1.



Рисунок П2.11 $-$ Построение переходной кривой.

Для этого сначала из палитры «Signal Producer» выбирается блок «const» (по умолчанию его значение равно 1). Затем из палитры «Linear System» блок «transferFunction». После того, как блок расположен на листе диаграммы щелчком правой кнопки мыши вызывается окно настроек этого блока (см. рис. 4.2.) и устанавливаются соответствующие значения.

Последний блок - блок графиков «plot» из палитры «Signal Consumer».

Расположенные блоки соединяются линиями функциональных связей. В настройках моделирования «Simulate -> Simulation Properties…» устанавливается время начала, конца моделирования и частота (соответственно: Start = 0, End = 20, Frequency = 5). Модель запускается на выполнение (командой «Simulation -> Go» или нажатием клавиши F5 клавиатуры или нажатием кнопки Go панели инструментов). Результат представлен на рис. 4.1.

Для распечатки переходных кривых на принтере следует увеличить окно графиков «plot» во весь экран и дать команду «File -> Print…» или нажать на кнопку «Print» панели инструментов.


Рисунок П2.12

 **4.2. Моделирование одноконтурной АСР.**

**Задание.** Построить модель одноконтурной АСР с объектом, заданным передаточной функцией вида

.

и ПИ-регулятором с настройками К0 = 0,08 и К1 = 0,1.

Построить переходные характеристики по заданию и по ошибке.

**Решение.** Строится модель, представленная на рис. 4.3.


Рисунок П2.13

Для моделирования объекта управления были использованы блоки передаточной функции (transferFunction), запаздывания (timeDelay) и константы (const).

Сигнал задания определяется в блоке константы (const). Выходные сигналы регистрируются в блоке графика (plot).

Настройки ПИ-регулятора задаются усилительными блоками (gain). Также в модели регулятора использован интегратор (integrator) и сумматор (summingJunction).

Обратная связь реализована с помощью формирователя связи (wirePositioner) и сумматора (summingJunction). По умолчанию все сигналы, поступающие на вход сумматора, берутся с положительными знаками. Для смены знака на противоположный (например, для организации отрицательной обратной связи) необходимо нажать клавишу «Ctrl» и, удерживая ее в нажатом положении, щелкнуть правой кнопкой мыши по соответствующему входу.

После расположения блоков на диаграмме, соединения их линиями связи и настройки диаграмма запускается на исполнение (см. рис. 4.3.).

**5. СТАНДАРТНЫЕ БЛОКИ ПАКЕТА ПРОГРАММ VISSIM**

Пакет программ VISSIM предназначен для моделирования процессов в системах

управления объектами, которые можно описать дифференциальными уравнениями или передаточными функциями. Работа пакета программ VISSIM описана в [1]. Данный пакет предоставляет исследователю богатую библиотеку различного рода блоков структурных схем и вспомогательных блоков (blocks) для моделирования динамических систем. Ниже подробно описаны функции блоков, которые предоставляет данный пакет программ.

**1.1. Блоки аннотаций (Annotation)**

Пакет VISSIM содержит следующие блоки аннотаций:

- Comment - блок комментария;

- Date - блок текущей даты и времени;

- ScalarToVec - группировка в одну линию нескольких соединительных линий;

- Variable - блок переменной;

- WireLabel - однострочный комментарий;

- VecToScalar - расщепление группового соединения;

- WirePositioner - блок позиционирования соединений на экране для улучшения

наглядности.

1. Annotation/comment - блок комментария.

Добавляет окно комментария к схеме. Для ввода текста в окно комментария

щелкните ПК мыши на блоке. Курсор мыши изменится на I-знак (мигающий курсор в наборном поле блока), показывающий, что блок находится в режиме вставки. Для выхода из режима вставки снова щелкните ПК мыши на блоке.

2. Annotation/date - блок даты.

Отображает текущую дату и время в формате: день, номер месяца, ч, мин, с, год. Для корректировки времени или даты перейдите в System Control Panel. Этот блок не имеет параметров.

3. Annotation/scalarToVec - блок объединения сигналов.

Объединяет входные сигналы в одну векторную связь. Используйте Edit/Add Input и Edit/Remove

Input для изменения числа входных меток. По умолчанию - три. Этот блок не имеет параметров.

4. Annotation/variable - блок переноса переменной.

Передает значения данных через схему без гибких связей. Блоки имеют названия совместно используемых сигналов. Только один блок, давший имя переменной, может иметь вход, но блоки переменной могут иметь несколько выходов. Переменная определяет имя блока. Если имя переменной начинается двоеточием (:), то это локальная переменная, на которую можно ссылаться только на текущем уровне схемы, что позволяет вам использовать блоки с одинаковым u1080 именем в различных составных блоках.

Имеется три специальных имени переменных, встроенных в VISSIM:

- $firstPass - генерирует начальный единичный импульс на первом шаге моделирования;

- $lastPass - генерирует заключительный единичный импульс на последнем шаге

моделирования;

- $runCount - содержит счетчик количества операций при многократном выполнении

моделирования. Используйте эту переменную с параметром автоматического запуска

из Simulate/Change Parameters... для выполнения моделирования методом Monte Carlo

и подстройки параметров.

5. Annotation/vecToScalar - блок разделения объединенных сигналов.

Разбивает векторную связь на скалярные выходные сигналы. Используйте команды Edit/Add Output и Edit/Remove Output для установления числа выходных меток. По умолчанию - три. Этот блок не имеет параметров.

6. Annotation/wireLabel - текстовый комментарий на поле модели.

Вставляет однострочный комментарий, соответствующий помеченной части схемы. Для установки или подавления обрамления используйте параметр Box WireLabels в диалоговом окне Edit/Preferences... WireLabel определяется комментарием.

7. Annotation/wirePositioner - блок для позиционирования проводов.

Позволяет располагать гибкие связи с минимальными нагромождениями на экране.

Блок состоит из входной и выходной меток, соединенных гибкой связью. Они не требуют дополнительного времени на вычисления во время моделирования. Этот блок не имеет параметров.

**1.2. Арифметические блоки (Arithmetic)**

Пакет VISSIM содержит следующие арифметические блоки:

 - 1/*х* - инверсия входной величины: выходной сигнал равен обратной величине

 входного: *y* = 1 *x* (не путать с блоком деления).

 -*х* - инвертор знака, знак выхода противоположен знаку входа: *y* = −*x* .

 \* - блок умножения, количество входов может быть любым: *y* = *x*1 ×*x*2 .

 / - блок деления имеет два входа, его выход равен отношению величин верхнего и нижнего входных сигналов: *y* = *x*1 *x*2 .

 abs - блок, вычисляющий абсолютное значение входного сигнала: *y* = *x* .

 gain - блок-усилитель с одним входом, выполняет умножение сигнала на константу, которая вводится в меню блока: *y* = *ax* .

  pow - блок с двумя входами возведения первого сигнала *x*1 в степень, равную величине второго сигнала *x*2 (второй сигнал может отсутствовать, в этом случае степень по умолчанию равна двум): 2 1 *y* = *x x* .

 sign - определяет знак входного сигнала: *y*=sign(*x*).

 SummingJunc4tion - сумматор, количество входов и знаки слагаемых можно

изменять: *y* = *x*1+ *x*2+....+ *x*N+*C*, имеет начальное смещение выхода *С*, которое

устанавливается в меню блока.

**1.3. Логические блоки (Boolean)**

Пакет VISSIM содержит следующие логические блоки с двумя входами и одним

выходом, который может принимать значения 0 или 1:

 > ( *x*1 больше, чем *x*2 );

 < ( *x*1 меньше, чем *x*2 );

 ≥ ( *x*1 больше или равно *x*2 );

 ≤ ( *x*1 меньше или равно *x*2 );

 == ( *x*1 равно *x*2 );

 != ( *x*1 не равно *x*2 );

 and - поразрядное “И” *x*1 и *x*2 ;

 or - поразрядное “ИЛИ” *x*1 и *x*2 ;

 xor - поразрядное исключительное “ИЛИ” *x*1 и *x*2 .

Пакет VISSIM содержит логический блок с одним входом и одним выходом, который может принимать значения 0 или 1:

not - Не, логическое отрицание.

Булевы (логические) блоки не имеют параметров. Нажатием ПК мыши на любом булевом блоке можно выбрать нужное отношение.

**1.4. Блоки интеграторов (Integration)**

Пакет VISSIM содержит следующие блоки интегрирования:

 - Integrator - простой интегратор;

 - LimitedIntegrator - интегратор с пределами интегрирования;

 - ResetIntegrator - интегратор со сбросом;

 - TransferFunction - передаточная функция.

В этих блоках параметр Initial Condition показывает начальное значение выходной координаты интегратора, которое по умолчанию равно нулю. Параметр ID представляет идентификационный номер блока. Он содержит порядковый номер, который VISSIM назначает интегратору. По умолчанию - 0, максимум - 255 для Personal VISSIM и 16344 для VISSIM.

1. Integration/integrator - простой интегратор.

*y* = ∫ *x* .

Выполняет численное интегрирование входной величины. Используйте команду

Simulate/Change Parameters... для выбора метода интегрирования.



2. Integration/limitedIntegrator - интегратор с пределами интегрирования.

Выполняет численное интегрирование входной величины, ограничивая внутренние

состояния, а следовательно, и выход с возможностью установления верхней и нижней

границ. Если значение интеграла достигает предела, то, как только производная изменит знак, интегрирование идет в обратную сторону от предела. Используйте команду Simulate/Change Parameters... для выбора метода интегрирования. Этот блок имеет три входа: *x*1 - производная, *x*2 - верхний предел, *x*3 - нижний предел.



3. Integration/resetIntegrator (1/*S*) - интегратор со сбросом, выходной сигнал которого описывается следующим алгоритмом: если *x*2 < 1 , то *y* = ∫ *x*1; иначе *y* = *x*3 .

Блок выполняет численное интегрирование входной величины с возможностью

сброса. Используйте команду Simulate/Change Parameters... для выбора метода

интегрирования. Параметр Initial Condition может быть заблокирован, если на входе *x*2 - не логический 0 на первом шаге моделирования.



1. Integration/transferFunction - передаточная функция.

Параметр 1: представляет передаточную функцию, определенную через полиномы

числителя и знаменателя. Параметр Gain - указывает коэффициент передачи. VISSIM

корректирует коэффициент передачи так, чтобы коэффициенты многочленов числителя и знаменателя при старших степенях равнялись единице. Окно Numerator служит для ввода коэффициентов многочлена числителя. Окно Denominator - для ввода коэффициентов многочлена знаменателя. Введите коэффициенты, отделяя их пробелами, начиная с самого высокого порядка. Не включайте знаки препинания или *s*-оператор. VISSIM определяет порядок передаточной функции по числу коэффициентов в знаменателе. Например, передаточная функция *n*-го порядка будет иметь *n*+1 коэффициентов знаменателя. Параметр Discrete - определяет дискретную передаточную функцию *Z*-области. По умолчанию - непрерывная *s*-область. Параметр *dT* определяет время для дискретной передаточной функции.

Параметр 2: представляет передаточную функцию, определенную как .MAT-файл. Когда блок читает .MAT-файл, то переводит матрицы пространства состояний A, B, C, D в SISO передаточную функцию *W*(*s*).

При моделировании блок передаточной функции использует эти SISO передаточные функции *W*(*s*). VISSIM поддерживает только SISO определения. Use .MAT File показывает, что VISSIM будет выполнять передаточную функцию, определенную в .MAT-файле. Параметр MAT file определяет имя файла, который используется как вход для блока передаточной функции. Наберите имя файла в текстовом окне или

щелкните ПК мыши на <Open. MAT File> для выбора из списка существующих файлов.

Ограничения: VISSIM поддерживает передаточные функции u1076 до 45-го порядка.

Personal VISSIM поддерживает передаточные функции до 15-го порядка.

**1.5. Блоки нелинейности (Nonlinear)**

Пакет VISSIM содержит следующие блоки нелинейностей:

- CROSSDETECT - детектор пересечения сигналом заданного уровня;

- DEADBAND - зона нечувствительности;

- INT - округление (усечение до целого);

- LIMIT - блок ограничений заданными пределами;

- MAX - максимум входных сигналов;

- MERGE - условный выбор;

- MIN - минимум входных сигналов;

- QUANTIZE - квантователь входного сигнала;

- RELAY - оператор реле с двумя состояниями;

- SAMPLEHOLD - синхронизированная защелка входного сигнала;

- MAP - одно- или двухмерные кусочно-линейные отображения.

1.  Nonlinear/crossDetect - детектор увеличения или уменьшения.

Выходная величина *y* = 1, если *x* пересекает величину, установленную в Cross Point,

с положительным коэффициентом наклона; *y* = -1, если *x* пересекает величину,

установленную в Cross Point, с отрицательным коэффициентом наклона; *y* = 0 в

противном случае. По умолчанию величина в Cross Point равна 0.

2. Nonlinear/deadband - зона нечувствительности (мертвая зона).

Выдает выходной сигнал, равный входному сигналу, уменьшенному на зону "потери воздействия", где сигнал равен нулю. Этот блок используется для моделирования люфта в механических системах, таких как зубчатая передача или цепной механизм. Параметр Dead Band показывает ширину зоны нечувствительности. По умолчанию – 0,2. Выходной сигнал равен 0 в центре зоны.

3.  Nonlinear/int - блок округления чисел с плавающей точкой до целых. Выходной сигнал равен целой части числа на входе (*y* = integer part *x*). Этот блок не имеет параметров.

4. Nonlinear/limit - блок насыщения (ограничитель уровня).

Если *x* меньше нижней границы, то *y* равен нижней границе, иначе если *x* больше верхней границы, то *y* равен верхней границе, иначе *y* равен *x*.

Блок ограничивает выходной сигнал в установленных верхнем и нижнем пределах. Параметр Lower Bound - наименьшая величина, которую может достигать выходной сигнал. По умолчанию - (-100). Параметр Upper Bound - наибольшая величина, которую может достигать выходной сигнал. По умолчанию - 100.

5. Nonlinear/map - кусочно-линейная нелинейность из .MAP-файла.

*y* = lookup (*x*) or *y* = lookup ( *x*1 , *x*2 ).

Задает одномерное или двухмерное кусочно-линейное отображение независимой переменной в одну или более зависимых переменных. Используйте блок вывода во внешний файл или любой редактор файла для создания VisSim-файлов отображения. Зависимые переменные - линейная интерполяция значений независимых переменныхмежду точками отображения и экстраполяция для значений вне пределов таблицы. Это свойство может использоваться для статической функциональной аппроксимации измеряемых данных или для калибровки устройства. Параметр Map File Name указывает имя многостолбцового файла, который сохраняет данные в текстовом формате ASCII.

Нажмите кнопку <Open New File... >, чтобы выбрать нужный файл из списка

существующих. Нажмите кнопку <Browse Data...> для чтения и редактирования

выбранного файла отображения. Параметр Map Dimensions указывает размеры матрицы и правило формирования зависимой переменной (*y*). При формировании зависимой переменной по правилу 1-D Mapping сопоставляет одну или более зависимых переменных одной независимой переменной. Файл данных должен располагаться так, чтобы в первом столбце чисел было множество независимых переменных в линейно-увеличивающемся или уменьшающемся порядке, а каждый последующий столбец соответствовал зависимой переменной и каждая строка отображала независимую переменную на зависимую переменную. Используйте команду Edit/Add Output, чтобы добавить соединительные метки для каждой зависимой переменной в блоке отображения. Самая верхняя соединительная метка вывода соответствует крайнему левому столбцу зависимых переменных в таблице. Правило формирования 2-D Mapping позволяет ставить в соответствие двум независимым переменным одну зависимую переменную. Соответствие ставится следующим образом: значения первой независимой переменой записываются в первой строке из второй колонки через пробел, а значения второй независимой переменной записываются из второй строки в первой колонке. Получается, что символ, находящийся на пересечении первой строки и первого столбца – пробел. На пересечении значений независимых переменных ставится значение зависимой переменной. Примером может u1089 служить таблица умножения Пифагора. Одномерные отображения позволяют задать до 8000 строк данных. Двухмерные

отображения позволяют задать до 89x89 элементов данных (90х90 матрицу).



6. Nonlinear/max - блок выделения максимального сигнала или числа.

*y* = maximum ( *x*1 , *x*2 ).

Выход - наибольшее значение из входных величин. Этот блок не имеет параметров.

7.  Nonlinear/merge - блок переключения входных сигналов.

Выходной сигнал принимает значение *x*2 , если *x*1 ≥ 1, и *x*3 - в противном случае.

Этот блок не имеет параметров.



8. Nonlinear/min - блок выделения минимального сигнала или числа.

*y* = minimum ( *x*1 , *x*3 ).

Выход - наименьшее значение входной величины. Этот блок не имеет параметров.

9. Nonlinear/quantize - блок округления или квантования по уровню.

Усекает или округляет входную величину до ближайшего числа, кратного

разрешающей способности (точности), в зависимости от знака входа и разрешающей способности. Если вход и разрешающая способность имеют тот же самый знак, тогда сигнал усекается по величине. Если вход и разрешающая способность имеют противоположные знаки, то сигнал округляется по величине до следующего кратного разрешающей способности. Это лучше всего поясняется примером, приведенным в таблице.



Resolution определяет значение квантователя. По умолчанию – 0,05.

10.  Nonlinear/relay - трехуровневое реле.

Моделирует блок-реле с тремя состояниями (-1, 0, 1). Параметр Dead Band

показывает ширину зоны нечувствительности относительно нулевого значения входного сигнала и таким образом создается оператор-реле с тремя состояниями (-1, 0, 1). Зона нечувствительности должна быть положительна, ее значение по умолчанию – 0,2. Если зона нечувствительности равна 0, то моделируется реле с двумя состояниями (-1, 1).

11.  Nonlinear/sampleHold - управляемый переключатель (включатель) фиксатор. Выходной сигнал описывается следующим алгоритмом: если *x*1 ≥ 1, то *y* = *x*2 , иначе *y* не изменяется. Блок фиксирует входную величину по управлению тактового сигнала. Параметр Initial Condition определяет начальное значение для *y*. По умолчанию - ноль.

**1.6. Генераторы шумов (Random generators)**

Описание блоков генераторов шумов (случайных чисел) – Random Generators.

1.  Random Generators/gaussian – нормальный гауссовский шум.

Этот блок генерирует нормально распределенный шум. Параметр Mean показывает центр распределения шума. По умолчанию – ноль. Параметр Standard Deviation показывает расстояние от среднего значения (среднее значение отклонения - девиация), которое занимает одна стандартная девиация. По умолчанию – единица. Начальное число случайной последовательности задается в диалоговом окне Simulate/Change Parameters…

2. Random Generators/uniform - произвольный однородный шум.

Этот блок однородно распределяет произвольный шум со значениями между 0 и 1. Параметр Time Delay (sec) определяет временную задержку перед вычислением величины шума. По умолчанию - ноль. Начальное число случайной последовательности задается в диалоговом окне Simulate/Change Parameters...

**1.7. Блоки-задатчики времени (Real time)**

1.  Real time/rt-DataIn - блок считывания данных в реальном времени.

Считывает данные в реальном времени с аналогово-цифровой платы расширения компьютера. Заметим, что этот функциональный блок доступен только с приобретением расширения VISSIM/RT. Блок ввода данных в реальном времени содержит следующую информацию: заголовок; канал; разрешение канала; класс канала; тип канала.

2. Real time/rt-DataOut - блок записи данных через АЦП-плату в ПК.

Записывает данные в реальном времени на аналогово-цифровую плату расширения компьютера. Заметим, что этот функциональный блок доступен только с приобретением расширения VISSIM/RT. Блок ввода данных в реальном времени содержит следующую информацию: заголовок; канал; тип канала.

**1.8. Блоки-получатели сигналов (Signal consumers)**

Пакет VISSIM содержит следующие блоки-получатели (или регистраторы, или приемники) сигналов:

- CONSTRAINT - ограничение для статического уравнения;

- DISPLAY - цифровой вывод сигнала на экран;

- ERROR - флаг ошибки;

- EXPORT - экспорт, запись сигналов в файл данных;

- METER - вывод сигнала на стрелочное показывающее устройство типа вольтметра, изображенного на экране;

- PLOT - устройство вывода до четырех графиков выходных сигналов модели;

- STOP - блок условного останова моделирования.

1. Signal Consumers/constraint - блок указания точности решения алгебраического уравнения. Ограничение для алгебраического уравнения используется вместе с "неопределенными" блоками. Раздел меню Tolerance определяет точность решения. Чем меньше допуск (ошибка), тем больше время вычисления. По умолчанию задан допуск 0,00001. Когда VISSIM обнаруживает блоки ограничения, то спрашивает, нужно ли активизировать итерацию Ньютона-Рафсона. Итерация Ньютона-Рафсона решает уравнение получением значений для "неопределенных" блоков, которые заставляют блоки ограничения стремиться к нулю насколько это возможно. Используйте команду Simulate/Change Parameters...., чтобы установить максимальное количество итераций, допустимую ошибку и возмущение для итерации Ньютона-Рафсона. Приложения включают неявные системы и подстройку системы.

2. Signal Consumers/display - блок цифрового указателя сигнала.

Отображает текущую величину входного сигнала с шестью или пятнадцатью значащими цифрами. По умолчанию - шесть значащих цифр. Используйте команду Edit/Preferences... для установки высокой точности отображения. Этот блок не имеет параметров.

3.  Signal Consumers/error - блок остановки моделирования при *x* ≠ 0 .

Сообщает об ошибке в моделировании. Когда входной сигнал становится

ненулевым, блок ошибки и все составные блоки, содержащие его, высвечиваются

красным цветом и моделирование останавливается. Чтобы сбросить состояние ошибки, щелкните ПК мыши на блоке ошибки. Этот блок не имеет параметров.



4. Signal Consumers/export - блок записи данных в файл.

Записывает сигналы в файл данных формата ASCII. Используйте команду Edit/Add Input и Edit/ Remove Input для установки числа экспортируемых сигналов. По умолчанию - три, максимум - шестнадцать. Окно Data File Name определяет файл экспорта. Нажмите кнопку <Open New File...>для выбора файла из списка существующих файлов данных. Нажмите кнопку <Browse Data...> для чтения и редактирования выбранного файла данных. По умолчанию VISSIM записывает файл в ваш текущий каталог, используя имя файла диаграммы с расширением .DAT. Параметры окна Data Point Time Delta определяют после того, как VISSIM запишет временные интервалы информации в файл данных. По умолчанию - Fixed Interval. Выбирайте Fixed Interval, когда отметки данных расположены через фиксированные интервалы. Step Size по умолчанию определяется параметрами моделирования. Блок экспорта не осуществляет интерполяцию. Если вы определили интервал, отличный от размера шага моделирования, данные будут экспортироваться с интервалами, кратными целому числу шагов моделирования. Заметим, что эта автоматическая корректировка фактически невидима, потому что размер шага экспорта ни модифицируется в диалоговом окне экспорта, ни отражается в заголовке файла данных (см. ниже). Если вы импортируете этот файл данных в другой процесс моделирования, вы должны изменить заголовок файла, чтобы отразить истинный интервал экспорта. Блок импорта будет осуществлять интерполяцию, обеспечивающую сохранение этого выбора времени. Выберите Time Data Column, когда отметки данных происходят в нерегулярные временные интервалы. Некоторые прикладные программы не могут читать файлы данных без столбцов данных времени. В этом случае вы должны выбрать Time Data Column, даже если отметки данных происходят в фиксированных интервалах. Допустимые столбцы - от 1 до 16 включительно. Окно Data File Info содержит следующую информацию: параметры Start Time и End Time представляют диапазон времени для записываемых данных. Они только для чтения, получаются из параметров моделирования. Окно Data Point Count определяет максимальное число данных, записываемых в файл данных. Каждая отметка данных занимает 8 байтов памяти. Максимально возможное число элементов данных - 128000. По умолчанию - 512 элементов данных, что требует 4096 байтов памяти на каждый вход.

Если выбран параметр Periodic Data Flush, то буфер экспорта будет отключен на

промежутке времени, определяемом пользователем и устанавливаемом в окне Flush Interval. Параметр Suppress VisSim Header полезен в случае, если файл данных необходимо импортировать в программу другого типа, отличного от VISSIM.

Информация заголовка определяется из значений в диалоговом окне экспорта. В нем записываются данные о временных интервалах для файла с использованием следующего формата: Fixed Interval #I=start time, end time, increment Variable Interval #T=number time column. Параметр Append To File позволяет добавить экспортируемые данные к существующим файлам при каждом выполнении моделирования, вместо того чтобы перезаписывать файл в начале каждого нового выполнения. Это полезно для многократно выполняемых приложений типа сбора данных, моделирования Монте Карло и обучаемой нейросети. Параметр Digits of Precision указывает значение точности для экспортируемых данных. По умолчанию - 15 значащих цифр.



5 . Signal Consumers/meter - стрелочный измеритель сигнала типа вольтметра.

Масштабирование и промежуточная градуировка выбираются автоматически, исходя из установленных в меню верхней и нижней границ. Lower Bound - наименьшая величина, отображаемая прибором. По умолчанию -(-10). Upper Bound - наибольшая величина, отображаемая прибором. По умолчанию - 10.



6. Signal Consumers/plot - графопостроитель регистрируемых сигналов.

Одновременно рисует графики до четырех входных сигналов в двумерной координатной сетке.

Используйте команды Edit/Add Input и Edit/Remove Input, чтобы изменить число входных меток на графическом блоке. Не имеется никаких ограничений на число или расположение графических блоков в вашей схеме. Щелкните ПК мыши на графическом блоке для обращения к Parameter Dialog Box (меню блока), который содержит все параметры, доступные вам. Параметр Fixed Bounds определяет, изменяет ли VISSIM границы графической шкалы для отображения входного сигнала, выходящего за границы диапазона, или отсекает значения сигнала вне существующих графических пределов. По умолчанию этот параметр выключен, поэтому весь график обновляется каждый раз, когда необходимо изменить масштаб. Если вы завершаете моделирование с этим отключенным параметром, VISSIM сам установит пределы. Вы можете с помощью Fixed Bounds ускорять последующие выполнения. Fixed bounds позволяет Вам устанавливать верхнюю и нижнюю границы *X* и *Y*. Чтобы исследовать часть существующего графика, введите поддиапазон, представляющий интерес, и эта часть графика будет отображена в полной графической области. Параметры моделирования имеют старшинство над пределамиграфика по координате *x*.

Параметр Frequency Domain обеспечивает частотный энергетический спектр, используя быстрое преобразование Фурье (*FFT* алгоритмE0т). Если ваш график *FFT* имеет непредвиденные скачки, проверьте адекватность частоты выборки для получения точныхрезультатов в диалоговом окне Simulate/Change Parameters... Основываясь на размере шага и диапазоне, установите Max Plotted Points так, чтобы вы действительно рисовали график на каждом шаге. Параметр Point Plot определяет, выводится ли график как ряд точек или как непрерывная линия. По умолчанию этот параметр выключен, что создает график в виде линии. Параметр Max Plotted Points определяет точность и гладкость графика. При составлении графика нескольких сигналов на черно-белом дисплее или принтере вы можете уменьшить количество точек, увеличивая расстояние между ними, но сохраняя четкое изображение линий. Каждая координата данных потребляет 8 байтов памяти. Максимальное количество используемых точек графика - 128000 на входной сигнал. По умолчанию - 512, что требует 4096 байтов памяти на каждый вход. Параметр Over Plot определяет, уничтожается ли предыдущий график или сохраняется. Используйте его для сравнения вариантов из серии моделирования. По умолчанию этот параметр выключен, поэтому каждое моделирование начинается с новым графическим экраном. Включенный параметр Over Plot расходует память. Параметр Plot Count определяет разрешенное количество последовательно выводимых графиков. По умолчанию - четыре, что требует четыре раза по 4096 байтов памяти для каждого входа. Параметр Geometric Markers позволяет идентифицировать сигналы с использованием квадратиков, ромбов, кругов и треугольников для графиков каждого сигнала. По умолчанию Geometric Markers не используются. Параметр Marker Count определяет число маркеров для различных графиков. По умолчанию - десять.

Параметр Grid Lines определяет, появляется ли координатная сетка на графиках. Координатная сетка помогает определять графические координаты. По умолчанию Grid Lines включены. Параметры Log *X* и Log *Y* определяют, будет ли ваш график линейный, логарифмический или полулогарифмический: по умолчанию устанавливается линейный. Для его получения не выбирайте никакого параметра. Для получения логарифмического выберите оба параметра. С целью получения полулогарифмического графика выберите или Log *X*, или Log *Y*. Заметим, что вы не можете изображать отрицательные величины на логарифмической оси. Любое отрицательное значение будет отсекаться за нижний предел шкалы.

Параметр *XY* Plot определяет, отображаются ли графики в режиме *XY* или во

временной области независимо от масштабирования оси. Когда действует *XY* Plot, верхний сигнал представляет *x*-ось, а нижний сигнал представляет *y*-ось. Отключите *XY* Plot, чтобы установить режим временной области. Тогда VISSIM будет рисовать всесигналы на *y*-оси, а время на *x-*оси. По умолчанию этот параметр выключен.

Параметр *X* Axis позволяет вам определять входной сигнал, который нужно

использовать для *x*-оси. Когда этот параметр выключен, по умолчанию этим сигналом считается первый из четырех входных сигналов на графическом блоке.

Параметр Title устанавливает заголовок графика до 80 символов. Графики первоначально идут без заглавия. Параметр Subtitle устанавливает подзаголовок графика до 80 символов только для копирования на принтер. Первоначально графики не имеют подзаголовка. Параметры *X* Label и *Y* Label устанавливают метки оси до 80 символов. По умолчанию *y*-ось не помечена. Метка по умолчанию для *x*-оси – Time, с. В *XY* графике метка для *x*-оси является маркировкой входного сигнала, устанавливаемой с помощью кнопки <Label Signals... >, которая вызывает диалоговое окно, куда вы можете вводить обозначение до 80 символов для каждого входного сигнала. Обозначения появляются в верхнем левом углу графика, с шаблоном линии, используемой для графика каждого сигнала. На цветном дисплее или устройстве печати метки имеют тот же цвет, что и соответствующие сигналы. Все входные сигналы первоначально не обозначены. Когда вы выбираете сигнал для *x*-оси *XY* графика, обозначение этого сигнала автоматически становится меткой для *x*-оси. Параметр Read Coordinates позволяет вам получать численные значения графических координат. Курсор мыши становится крестиком по размеру графика внутри графической области, при этом в левом нижнем углу будет появляться числовое значение координаты. Щелкните ЛК или ПК мыши, чтобы выйти из режима Read Coordinates. Для перемещения графического блока установите курсор мыши внутри графического экрана и установите его на новое место. Чтобы изменять размеры графического блока, используйте кнопки максимизации или минимизации, расположенные в верхнем правом углу графического блока, или нажмите ЛК мыши на границе графического блока и установите желаемый размер. Для получения печатной копии одиночного графика щелкните на пункте меню Control графического блока и выберите команду Print. VISSIM по умолчанию поставит график в очередь на системный принтер. VISSIM всегда выводит графики с подгонкой до полной страницы и не включает входные соединительные метки графического блока, минимизируемый/максимизируемый блок, или пункт меню Control.

7. Signal Consumers/stop - блок остановки моделирования при *x* ≥1.

Если *x* ≥ 2, то происходит остановка текущего выполнения. Если режим

автоперезапуска, то не производится следующий запуск. Иначе, если

*x* ≥ 1, тогда произойдет остановка текущего выполнения. Если режим автоперезапуска, то моделирование начинается сначала. Иначе, как обычно, останавливает моделирование, когда входной сигнал больше или равен единице. Этот блок не имеет параметров.

**1.9. Блоки задания сигналов (Signal producer)**

Пакет VISSIM содержит следующие блоки-генераторы сигналов:

- BUTTON – кнопка;

- CONST - константа, блок задания постоянного сигнала;

- IMPORT - импорт (чтение) данных из входного файла;

- PARABOLA - параболический сигнал;

- PULSETRAIN - периодическая последовательность нулей и единиц;

- RAMP - линейно нарастающий сигнал;

- REALTIME - реальное время в миллисекундах.

- SINUSOID – синусоидальная функция;

- SLIDER - управляемый мышью аналоговый сигнал, подобие потенциометра;

- STEP – функция-скачок;

- UNKNOWN - блок неизвестных, используется для решения алгебраических

уравнений в неявных системах.

Блоки (почти все) имеют только один выход.

1.  Signal Producers/button - кнопка для генерации скачка 1(*t*).

Если подвести курсор и нажать на кнопку, то *y* = 1 [Кнопка темная], если нажать еще

раз, то *y* = 0 [Кнопка светлая]. Позволяет вам динамически вставлять 0 и 1 во время

моделирования. Переключение цвета блока осуществляется щелчком ПК мыши на блоке. Имя кнопки изменяется командой Edit/Rename Block.... Этот блок не имеет параметров.

2.  Signal Producers/const - блок генерации постоянного числа (сигнала).

*y* = value. Генерирует постоянный сигнал. Value показывает значение величины (по умолчанию – один) выходного сигнала. Возможно использование π и ее констант. Величина сигнала устанавливается пользователем (после нажатия ПК мыши).



3. Signal Producers/import - блок воспроизводства данных из файла в

виде непрерывного сигнала. Записывает сигналы из ASCII файла данных. Используйте команды Edit/Add Input и Edit/ Remove Input для установки числа выходных сигналов. По умолчанию - три, максимум - шестнадцать. Массиву данных могут быть установлены начало, конец и значения приращения таким образом, чтобы импортируемые значения соотносились со временем моделирования. Значения данных будут линейно интерполироваться, если шаг по времени массива данных отличается от шага моделирования. Окно Data File Name определяет имя файла для импорта. Нажмите кнопку <Open New File...> для выбора файла из списка существующих файлов данных. Нажмите кнопку <Browse Data... > для чтения и

редактирования выбранного файла данных. Окно Data Point Time Delta указывает

интервал времени между отметками данных в файле. Если файл данных - файл пакета

VISSIM, то автоматически считывается информация о интервале времени из заголовка файла и соответственно устанавливаются параметры. По умолчанию - Fixed Interval. Если отметки данных находятся в фиксированных интервалах, то выбирайте Fixed Interval, иначе введите интервал в соответствующее текстовое окно. Выберите Time Data Column, когда отметки данных происходят в нерегулярные временные интервалы. Введите номер столбца, содержащего отметки данных времени, в соответствующее текстовое окно. Допустимые номера столбцов - от одного до шестнадцати. Окно Data File Info содержит разделы только для чтения Start Time и End Time, представляющие диапазон времени, в который записываются данные и, следовательно, диапазон времени, на котором допустимо моделирование. Эта информация получается из заголовка файла. Информация из заголовка сообщает VISSIM интервалы времени для данных файла в следующем формате: Fixed Interval #I=start time, End time, Increment Variable Interval #T=numberиtime column. Параметр Data Point Count определяет максимальное число элементов данных для считывания в VISSIM. VISSIM будет извлекать эту информацию из файла непосредственно. Максимально допустимое число элементов данных - 128000.

Каждый элемент данных потребляет 8 байт памяти на столбец.

4.  Signal Producers/parabola - блок квадратичного сигнала *y* = *a*(*t* − *c*)2 .

Создает параболический сигнал, исходя из времени моделирования. Параметр Time

Delay (*c*) определяет величину задержки в секундах (по умолчанию – ноль) перед

вычислением значения выходного сигнала. Параметр Slope Rate (*a*) масштабирует

кривизну параболы. Значение по умолчанию - один.

5.  Signal Producers/pulseTrain - блок - генератор импульсов.

Генерирует последовательность импульсов единичной амплитуды. Это полезно для синхронизации unitDelays и sampleHolds. Параметр Time Delay (секунд) определяет величину задержки перед вычислением значения выходного сигнала. Значение по умолчанию - 0. Параметр Time Between Pulses определяет время между импульсами. По умолчанию – 0,01. Вы можете добавить два входа к блоку pulseTrain, используя команду Edit/Add Input. Первый дополнительный вход позволяет вам вводить запаздывание извне, а второй - позволяет вам вводить извне время между импульсами. Эти дополнительные входы отменяют существующие параметры.

6.  Signal Producers/ramp - блок линейного сигнала *y* = *a*(*t* −задежки ). *y* = slope \* (time - time delay)= *a*( *t* − *t*задержки ), *y* = Фронт сигнала \* (текущее время – время задержки). Создает единичный пилообразный сигнал, исходя из времени моделирования. Параметр Time Delay ( *t*задержки ) - сдвиг во времени в секундах при вычислении значения выходного сигнала (по умолчанию – ноль). Параметр Slope (*а*) – фронт сигнала, по умолчанию - единица.

7.  Signal Producers/realTime - блок источника времени *bt* (*b*=10-3).

Выдает время в миллисекундах с начала вашего текущего сеанса VISSIM. Этот блок не имеет параметров. Обратите внимание, что это не время моделирования. Для получения времени моделирования выберите блок пилообразного сигнала.

8.  Signal Producers/sinusoid - генератор синусоиды *y* = *a*sin[ω(*t* − τ)]. Создает синусоидальный сигнал исходя из времени моделирования. Параметр Time Delay ( τ ) - сдвиг во времени в секундах u1087 при вычислении значения выходного сигнала. Его значение по умолчанию - ноль. Параметр Frequency ( ω) определяет частоту выходного синусоидального сигнала, определяется в рад/с. Значение по умолчанию - единица, что соответствует синусоиде с периодом 2π секунды. Параметр Amplitude ( *a* ) определяет максимальную величину выходного сигнала. Значение по умолчанию - единица.

9.  Signal Producers/slider - блок скользящего регулятора сигнала типа

 потенциометра. Позволяет с помощью мыши динамически изменять значение сигнала во время моделирования. Блок сдвига отображает текущую величину сигнала. Используйте горизонтальную линейку в блоке сдвига, чтобы корректировать величину сигнала между верхним и нижним пределом. Параметр Current Value определяет начальное значение выхода (по умолчанию – ноль). Параметр Upper Bound определяет максимальное значение выхода (по умолчанию – 100). Параметр Lower Bound определяет минимальное значение выхода (по умолчанию – 100).

10.  Signal Producers/step - блок источника сигнала типа скачка *y* = *a*1(*t* − τ) .

Если время меньше времени запаздывания Time delay ( τ ), то *y* = 0 , иначе *y* = *a* .

Создает единичный ступенчатый сигнал, исходя из времени моделирования. Параметр Amplitude (*a*) определяет максимальную величину выходного сигнала. Значение по умолчанию - единица. Параметр Time Delay ( τ ) - сдвиг во времени при вычислении значения выходного сигнала в секундах. Значение по умолчанию - ноль.

11. Signal Producers/unknown - блок начального задания при решениях неявных алгебраических уравнений. Применяется в неявных системах и в подстройке системы. "Неопределенности" должны быть всегда присоединены непосредственно или косвенно к блокам ограничения. VISSIM решает уравнение при численном воздействии на неопределенности, чтобы устремить ограничения к нулю. Максимальное количество итераций, допустимая ошибка и возмущение устанавливаются в команде Simulate/ Change Parameters... Этот блок не имеет параметров.

**1.10. Блоки задержки (Time delay)**

1.  Time Delay/Time Delay - блок динамической задержки.



Осуществляет задержку *x*2 в течение времени, определяемого *x*1 , где *s* – оператор Лапласа. Параметр Initial Condition устанавливает начальное условие для *y*. По умолчанию - ноль. Параметр Max Buffer Size управляет максимальным размером внутреннего буфера.

Блоку timeDelay требуется элемент буфера для каждого шага во время необходимойзадержки. Если timeDelay требует больше буферных элементов, чем позволяет Max BufferSize, отметки данных будут потеряны, порождая прерывистость, возникающую врезультате задержки выходного сигнала. Если сигнал - прерывистый, увеличьте величину Max Buffer Size. Для вычисления размера буфера определите максимальную требуемую задержку, разделите на шаг моделирования и добавьте 1. По умолчанию - 128 буферных элементов, а максимум - 8191. Каждый элемент занимает 8 байтов памяти.

2.  Time Delay/unitDelay - блок синхронизированной постоянной задержки.

Если *x*1 ≥ 1, то *y* = ybuffer, ybuffer = *x*2 , иначе *y* и ybuffer не изменяются. Определяет синхронизированное запаздывание для входных сигналов. Этот блок задает цифровую задержку в модели при непрерывном моделировании, обычно создается присоединением блока Signal Producers/pulseTrain к соединительной метке *x*1 блока unitDelay. Параметр Initial Condition устанавливает начальное значение для *y* (по умолчанию равное нулю).

**1.11. Блоки трансцендентных функций (Transcendental)**

Пакет VISSIM содержит следующие блоки трансцендентных функций с одним входом и одним выходом:

 ACOS – блок вычисления арккосинуса;

 ASIN – блок вычисления арксинуса;



 ATAN2 – блок вычисления арктангенса;

 BESSEL – блок вычисления функции Бесселя;

 COS – блок вычисления косинуса;

 COSH – блок вычисления гиперболического косинуса;

 EXP – блок вычисления экспоненты;

 LN – блок вычисления натурального логарифма;

 LOG10 – блок вычисления десятичного логарифма;

 SIN – блок вычисления синуса (блок-генератор синусоиды –SINUSOID, см.п.1.9);

 SINH – блок вычисления гиперболического синуса;

 SQRT – блок вычисления квадратного корня;

 TAN – блок вычисления тангенса;

 TANH – блок вычисления гиперболического тангенса.

Блоки осуществляют функциональное преобразование сигналов в пределах

допустимых значений для данных функций входных сигналов.

**1.12. Блок собственная функция-программа (Userfunction)**

Вызывает DLL функцию из вашей диаграммы. Для пользователей VISSIM в

основном каталоге VISSIM установлен файл README.TXT, содержащий описание

работы u1089 с DLL функциями. Эти файлы могут быть скомпилированы с использованием Microsoft C, Microsoft Fortran, and Borland C++. Ограничения: 255 входов, 16 выходов, 12 описательных параметров и сколько угодно произвольных параметров.