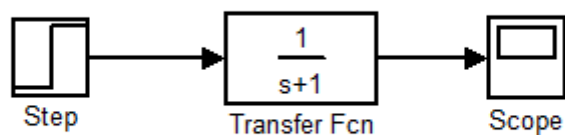


**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ
ФЕДЕРАЦИИ
КАЗАНСКИЙ (ПРИВОЛЖСКИЙ) ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ
НАБЕРЕЖНОЧЕЛНИНСКИЙ ИНСТИТУТ (ФИЛИАЛ)**

**Лабораторный практикум
«МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ В СРЕДЕ
SIMULINK»**



Набережные Челны

2015

УДК 681.3

ББК 32.973

Печатается по решению редакционно-издательского совета
Набережночелнинского института (филиала)
Казанского (Приволжского) федерального университета

Рецензенты: Марданшин Р. Г., канд. физ.-мат. наук, доцент

Демьянов Д. Н., канд. техн. наук, доцент

Ахметзянов И.З., Каримов В.С. Лабораторный практикум
«Математическое моделирование в среде Simulink». Набережные Челны: изд.-
полиграф. центр Набережночелнинского ин-та Казан. федер. ун-та, 2015. 104 с.

Лабораторный практикум является основным материалом для проведения лабораторных и практических занятий по дисциплине «Основы управления динамическими системами», «Практикум на ЭВМ». Он знакомит со средой имитационного моделирования динамических систем Simulink, и содержит лабораторные работы.

Лабораторный практикум разработан на кафедре "Системный анализ и информатика" и предназначен для организации и проведения лабораторных работ студентов по направлению подготовки 010400.62 "Прикладная математика и информатика" и направлению подготовки 220100.62 "Системный анализ и управление".

Илл. 37, табл. 12, список лит. 6 наим.

УДК 681.3

ББК 32.973

© Набережночелнинский институт КФУ, 2015

© И.З. Ахметзянов, В.С. Каримов 2015.

Содержание

1. Общие сведения о среде математического моделирования Simulink.....	5
2. Создание модели	7
2.1. Подготовка математического или логико-математического описания модели объекта и начало создания Simulink модели	7
2.2. Окно модели	12
3. Редактирование simulink-модели	15
3.1. Добавление текстовых надписей	15
3.3. Вставка блоков и их соединение	17
3.4. Форматирование объектов	22
4. Установка параметров расчета и его выполнение	24
4.1. Установка параметров расчета модели	25
4.2. Установка параметров обмена с рабочей областью	28
4.3. Установка параметров диагностирования модели	30
4.4. Выполнение расчета	31
5. Обзор библиотеки блоков Simulink	32
5.1. Sources - источники сигналов	32
5.2. Sinks - приемники сигналов	39
5.3. Continuous – аналоговые блоки	50
5.4. Discrete – дискретные блоки.....	52
5.5. Nonlinear - нелинейные блоки	54
5.6. Math – блоки математических операций.....	57
5.7. Signal&Systems - блоки преобразования сигналов и вспомогательные блоки	63
5.8. Subsystem – подсистемы.	67
Лабораторная работа №1	71
Лабораторная работа № 2	73
Лабораторная работа № 3	81

Лабораторная работа №4	90
Лабораторная работа №5	94
Литература	102

1. Общие сведения о среде математического моделирования

Simulink

Имитационное моделирование – метод, позволяющий строить модели, описывающие процессы так, как они проходили бы в действительности. **Динамическим** называется такой процесс, когда набор параметров некоторого объекта меняются **во времени** под влиянием внешних либо внутренних воздействий. Этот набор параметров еще называется **состоянием** объекта. Имитационное моделирование позволяет воспроизвести процессы в модели для различных начальных условий, внешних воздействий. Реалистичность результатов моделирования зависит от реалистичности используемой модели.

Simulink – это подсистема имитационного математического моделирования динамических процессов. Simulink является составной частью пакета Matlab и полностью интегрирован с ним.

Модели в Simulink состоят из набора графических блоков, представляющих компоненты объекта или какие-то функциональные элементы (источники сигналов различного вида, виртуальные регистрирующие приборы, средства анимации), и направленных связей между ними (линии распространения сигналов). Блоки различного назначения сгруппированы в библиотеки блоков (Blocksets) (аналогично пакетам прикладных программ). Каждый блок имеет набор настраиваемых параметров.

После создания модели запускается моделирование (simulating), воспроизводящее процессы во времени, происходящие в объекте. Этот этап в дальнейшем будет называться **симуляцией**.

При работе с **Simulink** пользователь имеет возможность модернизировать библиотечные блоки, создавать свои собственные, а также составлять новые библиотеки блоков.

При моделировании пользователь может выбирать метод решения дифференциальных уравнений, а также способ изменения модельного


времени (с фиксированным или переменным шагом). В ходе моделирования имеется возможность следить за процессами, происходящими в системе. Для этого используются специальные устройства наблюдения, входящие в состав библиотеки **Simulink**. Результаты моделирования могут быть представлены в виде графиков или таблиц.

2. Создание модели

На данном этапе составляется, собственно, математическая модель исследуемого объекта. Для этого используется соответствующий математический аппарат. Для описания динамических процессов чаще всего используются различные виды дифференциальных уравнений. Для корректного моделирования исследуемого объекта необходимы знания в соответствующей предметной области. Созданную математическую модель, возможно, придется адаптировать для ее реализации в системе Simulink с учетом имеющихся блоков.

2.1. Подготовка математического или логико-математического описания модели объекта и начало создания Simulink модели

Для создания модели в среде **Simulink** необходимо последовательно выполнить ряд действий:

1) Создать новый файл модели с помощью команды **File/New/Model**, или используя кнопку  на панели инструментов (здесь и далее, с помощью символа “/”, указаны пункты меню программы, которые необходимо последовательно выбрать для выполнения указанного действия). Вновь созданное окно модели показано на рисунке 2.1.

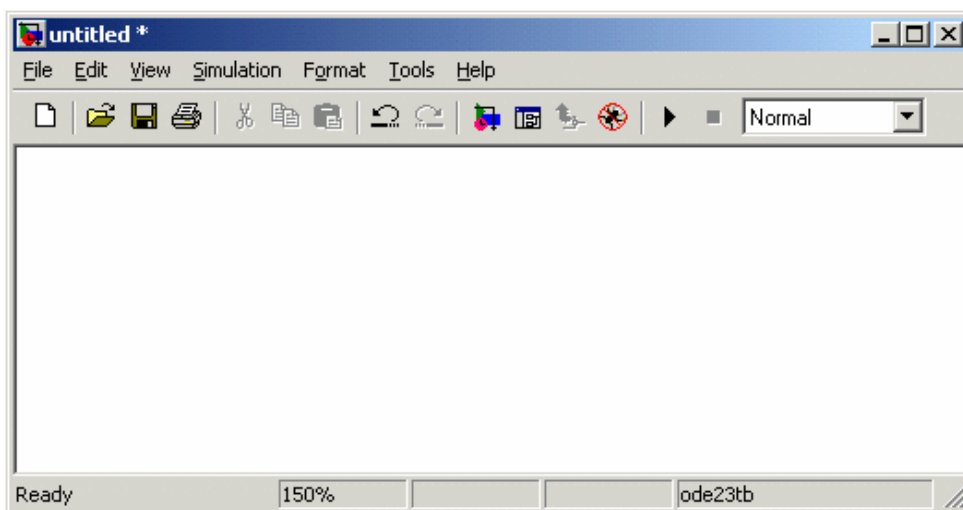


Рисунок 2.1 – Пустое окно модели

2) Создание модели в Simulink состоит в размещении в рабочей области графических блоков, реализующих элементы математической модели или выполняющие вспомогательные функции, и соединении их линиями распространения сигналов. Блоки Simulink сгруппированы в библиотеках блоков (Blocksets). Доступ к ним осуществляется с помощью средства Simulink Library Browser (рисунок 2.2).

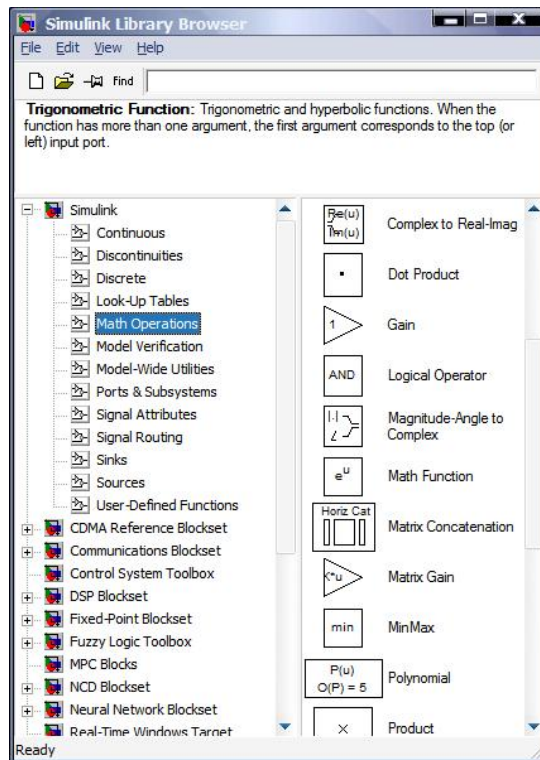


Рисунок 2.2 – Окно Simulink Library Browser

Для добавления блока в модель достаточно просто перетащить его мышкой в окно модели. Для блоков каждого вида существует определенный набор параметров, для настройки которых используется диалоговое окно.

На рисунке 2.3 показано окно модели, содержащее блоки.

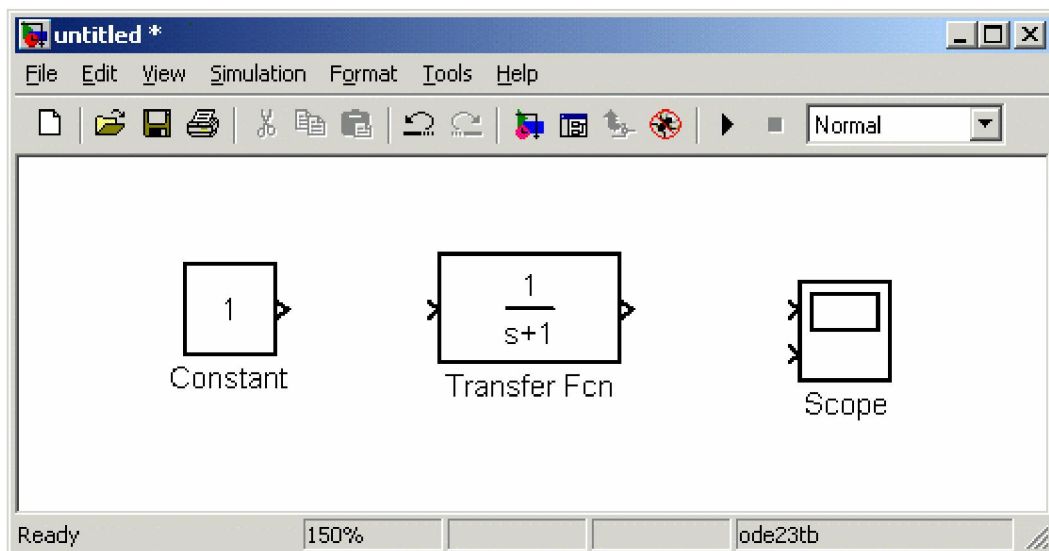


Рисунок 2.3 – Окно модели, содержащее блоки

Для манипуляций с блоками и сигнальными линиями в основном используется мышь и клавиатурные сокращения. Основные операции сведены в таблицу 1.

Таблица 1 – Основные операции с блоками и линиями

Манипуляция мышью или сочетание клавиш	Действие
Щелчок ЛКМ на блоке или линии	Выделение блока или линии
SHIFT + щелчок ЛКМ на блоке или линии	Добавление линии или блока к выделенным
Щелчок ПКМ на блоке или линии	Контекстное меню блока или линии
Перетаскивание блока с нажатой ЛКМ	Перемещение блока с сохранением соединения линий
Перетаскивание линии с нажатой ЛКМ	Перемещение линейного сегмента линии без нарушения связей
Перетаскивание блока с нажатой ПКМ Перетаскивание блока с нажатой ЛКМ + CTRL	Копирование блока и перемещение его
Перетаскивание линии с нажатой ПКМ	Разветвление линии из точки щелчка
Перетаскивание блока с нажатой ЛКМ + SHIFT	Перетаскивание блока с отсоединением линий
Выделение одного блока, затем щелчок ЛКМ + CTRL по другому блоку	Соединение выхода первого блока с первым свободным входом второго

Двойной щелчок ЛКМ на блоке	Вызов окна параметров блока
Двойной щелчок ЛКМ в пустом месте окна	Вставка текста в место щелчка
CRTL+C, CTRL + X, CTRL + V	Буфер обмена: Копирование, вырезание, вставка
CTRL + R	Вращение блока на 90°
CTRL + I	Отражение блока слева направо (сверху вниз)

Для удаления блока необходимо выбрать блок (указать курсором на его изображение и нажать левую клавишу мыши), а затем нажать клавишу **Delete** на клавиатуре.

Для изменения размеров блока требуется выбрать блок, установить курсор в один из углов блока и, нажав левую клавишу мыши, изменить размер блока (курсор при этом превратится в двухстороннюю стрелку).

3) Далее, если это требуется, нужно изменить параметры блока, установленные программой «по умолчанию». Для этого необходимо дважды щелкнуть левой клавишей мыши, указав курсором на изображение блока. Откроется окно редактирования параметров данного блока. При задании численных параметров следует иметь в виду, что в качестве десятичного разделителя должна использоваться точка, а не запятая. После внесения изменений нужно закрыть окно кнопкой **ОК**. На рисунке 2.4 в качестве примера показаны блок, моделирующий передаточную функцию и окно редактирования параметров данного блока.

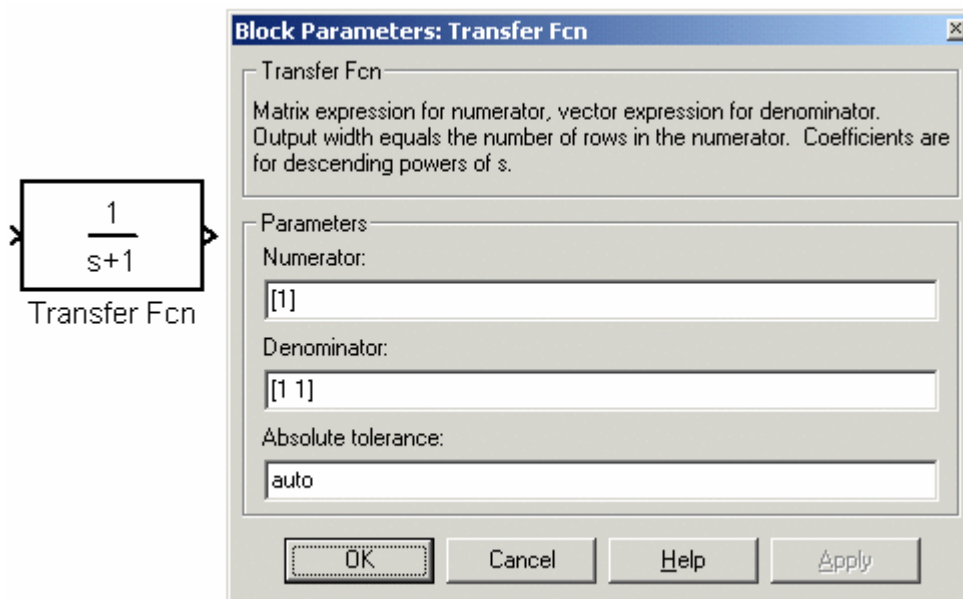


Рисунок 2.4 – Блок, моделирующий передаточную функцию и окно редактирования параметров блока

- 4) После установки на схеме всех блоков из требуемых библиотек, нужно выполнить соединение элементов схемы. Для соединения блоков необходимо указать курсором на «выход» блока, а затем, нажав и, не отпуская левую клавишу мыши, провести линию к входу другого блока. После чего отпустить клавишу. В случае правильного соединения, изображение стрелки на входе блока изменяет цвет. Для создания точки разветвления в соединительной линии нужно подвести курсор к предполагаемому узлу и, нажав *правую* клавишу мыши, протянуть линию.
- 5) После составления расчетной схемы необходимо сохранить ее в виде файла на диске, выбрав пункт меню **File/Save As...** в окне схемы и указав папку и имя файла. Следует иметь в виду, что имя файла не должно превышать 32 символа, должно начинаться с буквы и не может содержать символы кириллицы и спецсимволы. Это же требование относится и к пути файла (к тем папкам, в которых сохраняется файл). При последующем редактировании схемы можно пользоваться пунктом меню **File/Save**. При повторных запусках программы **Simulink** загрузка схемы осуществляется с помощью меню **File/Open...** в окне обозревателя

библиотеки или из основного окна **MATLAB**.

Для удаления линии требуется выбрать линию (так же, как это выполняется для блока) и нажать клавишу **Delete** на клавиатуре. Схема модели, в которой выполнены соединения между блоками, показана на рисунке 2.5.

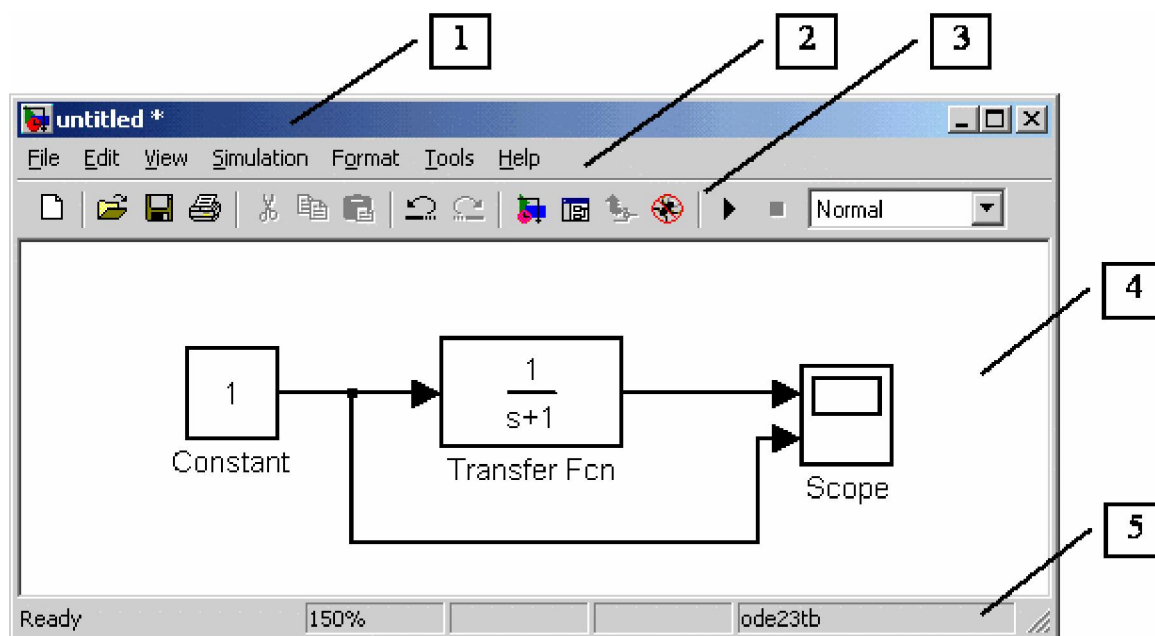


Рисунок 2.5 – Схема модели

2.2. Окно модели

Окно модели содержит следующие элементы (см. рисунок 2.5):

1. Заголовок с названием окна. (Вновь созданному окну присваивается имя **Untitled** с соответствующим номером)
2. Меню с командами **File**, **Edit**, **View** и т.д.
3. Панель инструментов.
4. Окно для создания схемы модели.
5. Строка состояния, содержащая информацию о текущем состоянии модели.

Меню окна содержит команды для редактирования модели, ее настройки и управления процессом расчета, работы файлами и т.п.:

- **File (Файл)** — Работа с файлами моделей.
- **Edit (Редактирование)** — Изменение модели и поиск блоков.

- **View (Вид)** — Управление показом элементов интерфейса.
- **Simulation (Моделирование)** — Задание настроек для моделирования и управления процессом расчета.
- **Format (Форматирование)** — Изменение внешнего вида блоков и модели в целом.
- **Tools (Инструментальные средства)** — Применение специальных средств для работы с моделью (отладчик, линейный анализ и т.п.)
- **Help (Справка)** — Вывод окон справочной системы.

Для работы с моделью можно также использовать кнопки на панели инструментов (рисунок 2.6).

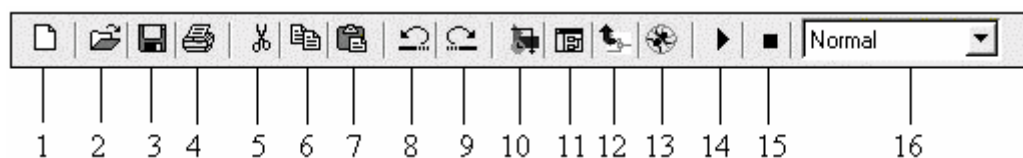


Рисунок 2.6 – Панель инструментов окна модели

Кнопки панели инструментов имеют следующее назначение:

1. **New Model** – Открыть новое (пустое) окно модели.
2. **Open Model** – Открыть существующий **mdl**-файл.
3. **Save Model** – Сохранить **mdl**-файл на диске.
4. **Print Model** – Вывод на печать блок-диаграммы модели.
5. **Cut** – Вырезать выделенную часть модели в буфер промежуточного хранения.
6. **Copy** – Скопировать выделенную часть модели в буфер промежуточного хранения.
7. **Paste** – Вставить в окно модели содержимое буфера промежуточного хранения.
8. **Undo** – Отменить предыдущую операцию редактирования.
9. **Redo** – Восстановить результат отмененной операции редактирования.
10. **Library Browser** – Открыть окно обозревателя библиотек.
11. **Toggle Model Browser** – Открыть окно обозревателя модели.

12. Go to parent system – Переход из подсистемы в систему высшего уровня иерархии («родительскую систему»). Команда доступна только если открыта подсистема.

13. Debug – Запуск отладчика модели.

14. Start/Pause/Continue Simulation – Запуск модели на исполнение (команда **Start**); после запуска модели на изображении кнопки выводится символ паузы (Приостановить моделирование); для возобновления моделирования следует щелкнуть по той же кнопке, поскольку в режиме паузы ей соответствует команда **Continue** (Продолжить).

15. Stop – Закончить моделирование. Кнопка становится доступной после начала моделирования, а также после выполнения команды **Pause**.

16. Normal/Accelerator – Обычный/Ускоренный режим расчета. Инструмент доступен, если установлено приложение **Simulink Performance Tool**.

В нижней части окна модели находится строка состояния, в которой отображаются краткие комментарии к кнопкам панели инструментов, а также к пунктам меню, когда указатель мыши находится над соответствующим элементом интерфейса. Это же текстовое поле используется и для индикации состояния **Simulink: Ready** (Готов) или **Running** (Выполнение).

В строке состояния отображаются также:

- масштаб отображения блок-диаграммы (в процентах, исходное значение равно 100%),
- индикатор степени завершенности сеанса моделирования (появляется после запуска модели),
- текущее значения модельного времени (выводится также только после запуска модели),
- используемый алгоритм расчета состояний модели (метод решения).

3. Редактирование simulink-модели

3.1. Добавление текстовых надписей

Для повышения наглядности модели удобно использовать текстовые надписи. Для создания надписи нужно указать мышью место надписи и дважды щелкнуть левой клавишей мыши. После этого появится прямоугольная рамка с курсором ввода. Аналогичным образом можно изменить и подписи к блокам моделей. На рисунке 3.1 показаны текстовая надпись и изменение надписи в блоке передаточной функции. Следует иметь в виду, что рассматриваемая версия программы (**Simulink 4**) не адаптирована к использованию кириллических шрифтов, и применение их может иметь самые разные последствия: отображение надписей в нечитаемом виде, обрезание надписей, сообщения об ошибках, а также невозможность открыть модель после ее сохранения. Поэтому, применение надписей на русском языке для текущей версии **Simulink** крайне нежелательно.

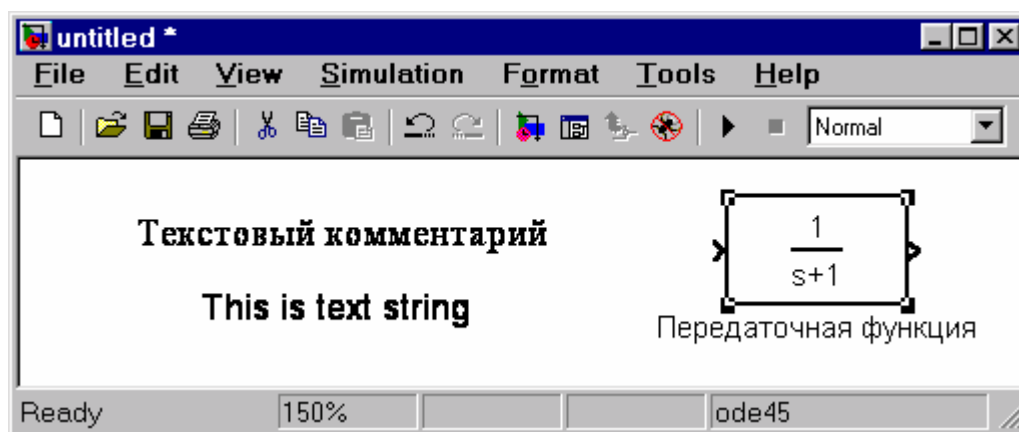


Рисунок 3.1 – Текстовая надпись и изменение надписи в **Transfer Function**

3.2. Выделение объектов

Для выполнения какого-либо действия с элементом модели (блоком, соединительной линией, надписью) этот элемент необходимо сначала выделить.

Выделение объектов проще всего осуществляется мышью. Для этого

необходимо установить курсор мыши на нужном объекте и щелкнуть левой клавишей. Произойдет выделение объекта. Об этом будут свидетельствовать маркеры по углам объекта (см. рисунок 3.1). Можно также выделить несколько объектов. Для этого надо установить курсор мыши вблизи группы объектов, нажать левую клавишу мыши и, не отпуская ее, начать перемещать мышь. Появится пунктирная рамка, размеры которой будут изменяться при перемещении мыши. Все охваченные рамкой объекты становятся выделенными. Выделить все объекты также можно, используя команду **Edit/Select All**.

Для стирания выделенного объекта можно вызвать команду **Clear** из меню **Edit** или из контекстного меню (рисунок 3.2). Контекстное меню очень удобно тем, что для любого объекта оно выводит перечень команд, которые доступны в данном состоянии.

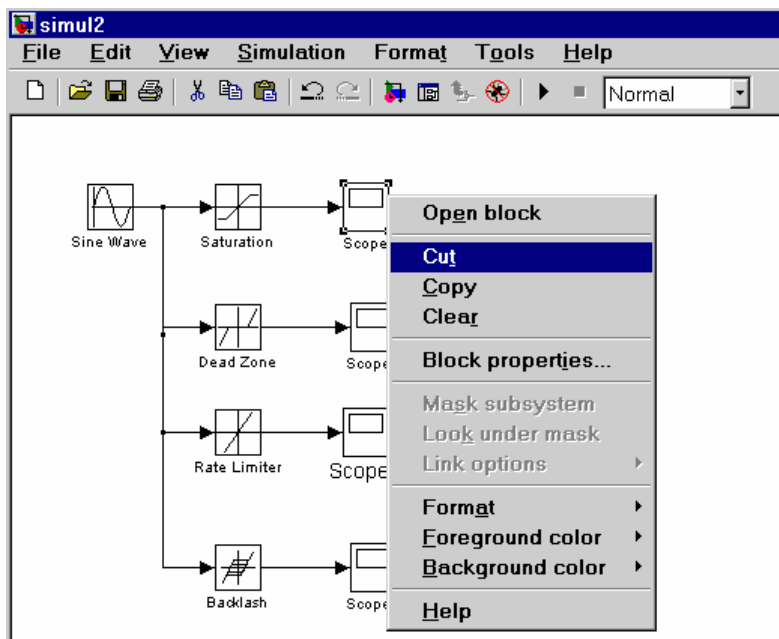


Рисунок 3.2 – Контекстное меню

Для восстановления объекта в окне модели следует щелкнуть левой кнопкой мыши в предполагаемом месте расположения объекта. После этого выполнение команды **Paste** из меню **File** окна **Simulink** или из контекстного меню помещает хранящийся в буфере объект (блок) в заданное место.

Следует учесть, что команда **Clear** стирает блок безвозвратно, то есть без

помещения его в буфер обмена. Однако эту операцию можно отменить командой меню **File / Undo** окна **Simulink**.

3.3. Вставка блоков и их соединение

Для подключения новых блоков нужны новые соединения. Они также легко выполняются с помощью мыши. Приемы ввода новых блоков и их соединений выполняются очень просто.

Блоки моделей обычно имеют входы и выходы. Как правило, выход какого-либо блока подключается к входу следующего блока и т. д. Для этого курсор мыши устанавливается на выходе блока, от которого должно исходить соединение. При этом курсор превращается в большой крестик из тонких линий. Держа нажатой левую кнопку мыши, надо плавно переместить курсор к входу следующего блока (рисунок 3.3), где курсор мыши приобретает вид крестика из тонких сдвоенных линий.



Рисунок 3.3 – Начало соединения блока источника с блоком осциллографа

Добившись протяжки линии к входу следующего блока, надо отпустить левую кнопку мыши. Соединение будет завершено и в конце его появится жирная стрелка. Щелчком мыши можно выделить соединение, признаком чего будут черные прямоугольники, расположенные в узловых точках соединительной линии (рисунок 3.4).

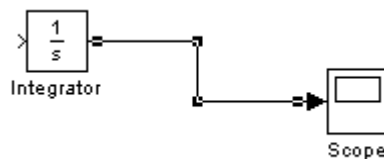


Рисунок 3.4 – Выделенное соединение

Создание отвода линии

Часто возникает необходимость сделать отвод от уже созданной линии. Пример создания такого отвода иллюстрирует рисунок 3.5. Заметим, что при нажатой клавише **Shift** отвод строится наклонными линиями.

В примере, показанном на рисунке 3.5, использована модель интегратора, подключенного к выходу источника прямоугольных импульсов. Чтобы можно было наблюдать осциллограммы как на выходе, источника, так и на выходе интегратора, в схему включен блок мультиплексора сигналов **Mux** с двумя входами. Чтобы подключить нижний вход к уже задействованному выходу источника, нам и понадобилось создать отвод линии.

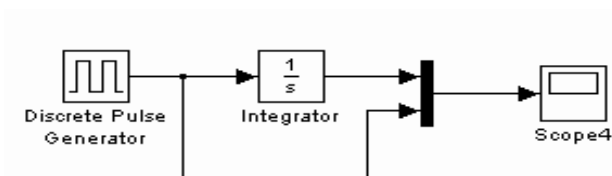


Рисунок 3.5 – Пример модели с отводом линии

Теперь можно запустить эту модель и посмотреть, какие сигналы действуют на выходах источник и интегратора. Результат запуска представлен на рисунок 3.6.

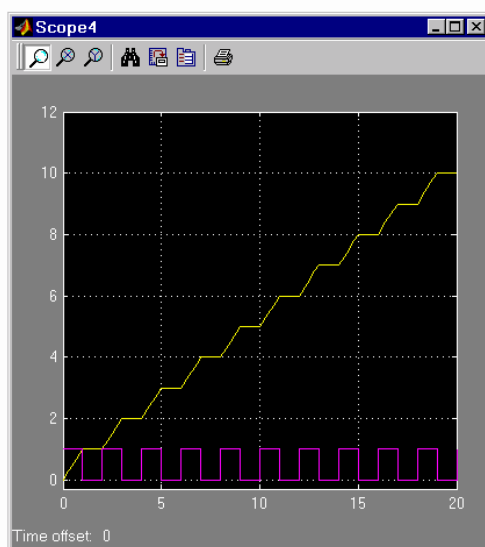


Рисунок 3.6 – Результат моделирования

Нетрудно убедиться в том, что сигнал на выходе интегратора представляет собой ступенчато нарастающую линию. Когда на выходе генератора имеется высокий (условно) уровень напряжения, на выходе интегратора сигнал линейно нарастает. Когда уровень на генераторе равен 0, сигнал на выходе интегратора остается неизменным. Этот пример дает наглядную иллюстрацию работы интегратора.

Удаление соединений

Для удаления соединительной линии достаточно выделить ее и выполнить команду **Clear** или **Cut**.

Изменение размеров блоков

Simulink имеет расширенные возможности редактирования блок-схем. Так, блоки в окне редактирования можно не только перемещать с помощью мыши, но и изменять в размерах. Для этого блок выделяется, после чего курсор мыши надо установить на кружки по углам блока. Как только курсор мыши превратится в двунаправленную диагональную стрелку, можно будет при нажатой левой кнопке растягивать блоки по диагонали, увеличивая или уменьшая их размеры (рисунок 3.7)



Рисунок 3.7 – Растяжение блока

Увеличенный в размерах блок показан на рисунок 3.8. Обратите внимание на то, что растягивается только графическое изображение (пиктограмма) блока, а размеры его названия в виде текстовой надписи не изменяются.

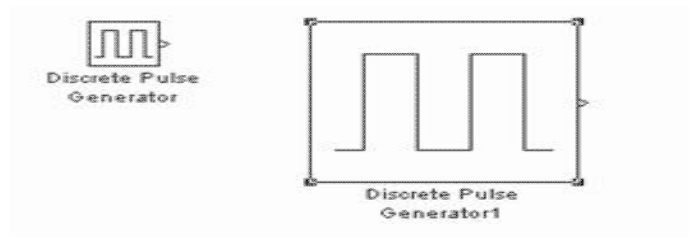


Рисунок 3.8 – Пример растяжения блока

Перемещение блоков и вставка блоков в соединение

Блок, участвующий в соединении, можно перемещать в окне модели, выделив его и перетаскивая, как обычно, мышью. При этом соединение не прерывается, а просто сокращается или увеличивается в длине. В длинное соединение можно вставить новый блок, не разрушая его и не выполняя сложных манипуляций. Рисунок 3.9 показывает вставку блока дифференцирующего устройства между источником синусоидального сигнала и осциллографом.

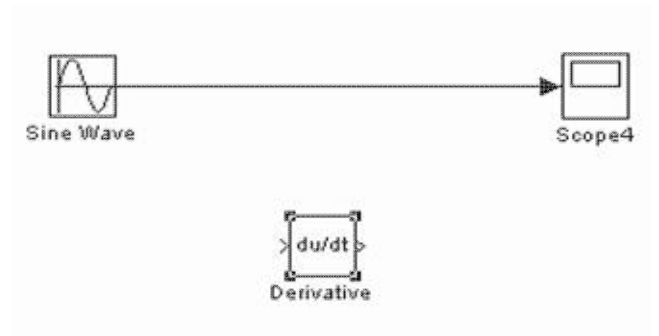


Рисунок 3.9 – Исходное соединение и блок для вставки

Результат вставки дифференцирующего устройства в соединение между источником и осциллографом показан на рисунке 3.10.

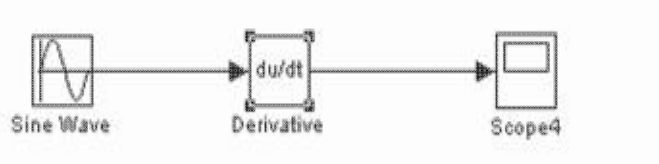


Рисунок 3.10 – Пример вставки блока в соединение

Однако подобная простая вставка возможна для блоков, имеющих один

вход и один выход, которые включаются в соединение. Попытка вставить таким образом мультиплексор будет безуспешной, поскольку он имеет два входа и не стыкуется с разрываемым соединением, чтобы вставить мультиплексор, следует удалить соединение между дифференцирующим устройством и входом осциллографа. Для этого соединение выделяется и выполняется команда **Edit/Clear**. После этого мультиплексор перемещается в нужное место и соединения создаются заново.

Моделирование дифференцирующего устройства

Итак, мы фактически создали модель дифференцирующего устройства и можем посмотреть, что происходит при дифференцировании синусоидального сигнала. Результат запуска созданной модели представлен на рисунке 3.11.

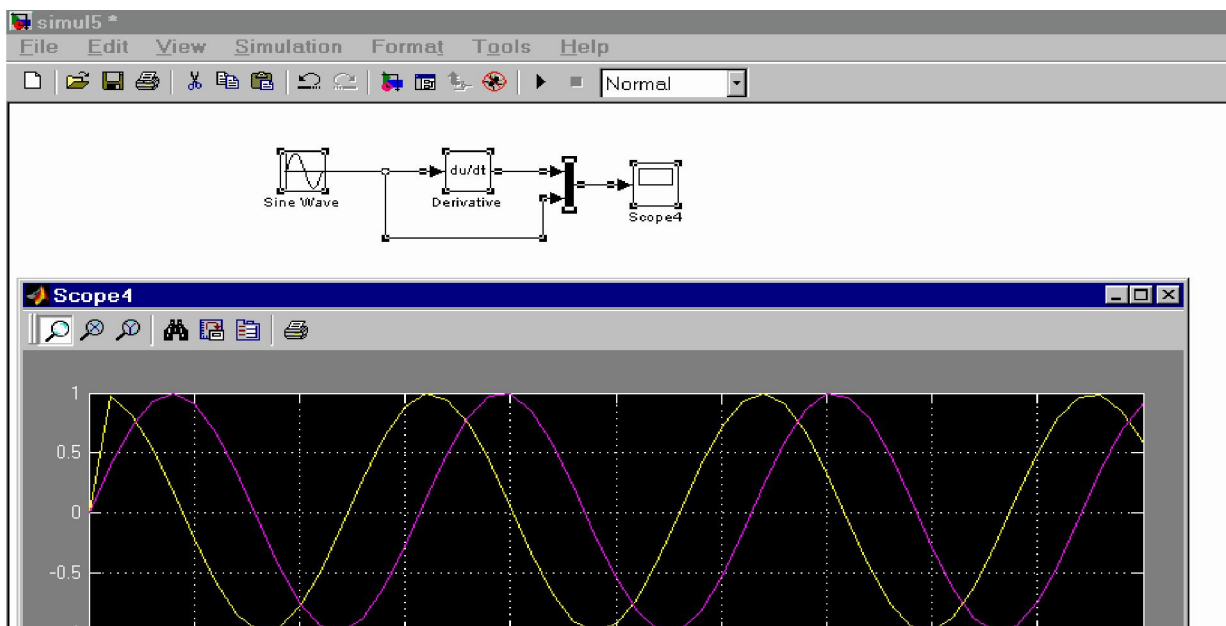


Рисунок 3.11 – Моделирование дифференцирующего устройства

Внимательно присмотревшись к осциллограммам, мы видим, что при входном синусоидальном сигнале выходной сигнал является косинусоидой. Это вполне отвечает математическим соотношениям для данного случая (как известно, производная $\sin(x)$ есть $\cos(x)$).

Однако в самом начале процесса дифференцирования хорошо виден изъян работы модели – при $t = 0$ производная равна не 1, а 0.

Это связано с тем, что процесс начинается при нулевых начальных условиях. Но довольно быстро ситуация исправляется, и в дальнейшем выходной сигнал становится косинусоидальным. Таким образом, дифференцирующее устройство можно использовать для точного сдвига на 90° гармонического сигнала любой частоты.

Команды Undo и Redo в окне модели

Большую помощь в редактировании оказывает команда **Undo** – отмена последней операции. Она поддерживает свыше ста различных операций, включая операции добавления и стирания линий. Эту команду можно вызвать с помощью кнопки в панели инструмента окна модели или из меню **Edit**. Для восстановления отмененной операции служит команда **Redo**.

3.4. Форматирование объектов

В меню **Format** (также как и в контекстном меню, вызываемом нажатием правой клавиши мыши на объекте) находится набор команд форматирования блоков.

Команды форматирования разделяются на несколько групп:

1. Изменение отображения надписей:

- **Font** — Форматирование шрифта надписей и текстовых блоков.
- **Text alignment** — Выравнивание текста в текстовых надписях.
- **Flip name** — Перемещение подписи блока.
- **Show/Hide name** — Отображение или скрытие подписи блока.

2. Изменение цветов отображения блоков:

- **Foreground color** — Выбор цвета линий для выделенных блоков.
- **Background color** — Выбор цвета фона выделенных блоков.
- **Screen color** — Выбор цвета фона для всего окна модели.

3. Изменение положения блока и его вида:

- **Flip block** – Зеркальное отображение относительно вертикальной оси симметрии.
- **Rotate block** – Поворот блока на 90^0 по часовой стрелке.
- **Show drop shadow** — Показ тени от блока.
- **Show port labels** — Показ меток портов.

4. Прочие установки:

- **Library link display** — Показ связей с библиотеками.
- **Sample time colors** — Выбор цвета блока индикации времени.
- **Wide nonscalar lines** — Увеличение/уменьшение ширины нескаллярных линий.
- **Signal dimensions** — Показ размерности сигналов.
- **Port data types** — Показ данных о типе портов.
- **Storage class** — Класс памяти (параметр, устанавливаемый при работе **Real-Time Workshop**).
- **Execution order** — Вывод порядкового номера блока в последовательности исполнения.

Команда **Format/Font** выводит окно с установками шрифта для текстовых надписей (рисунок 3.12).

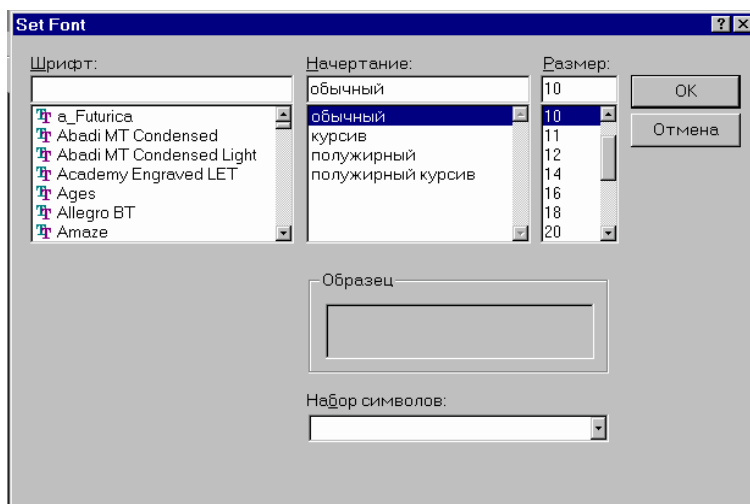


Рисунок 3.12 – Окно выбора шрифта

4. Установка параметров расчета и его выполнение

Перед выполнением расчетов необходимо предварительно задать параметры расчета. Задание параметров расчета выполняется в панели управления меню **Simulation/Parameters**. Вид панели управления приведен на рисунке 4.1.

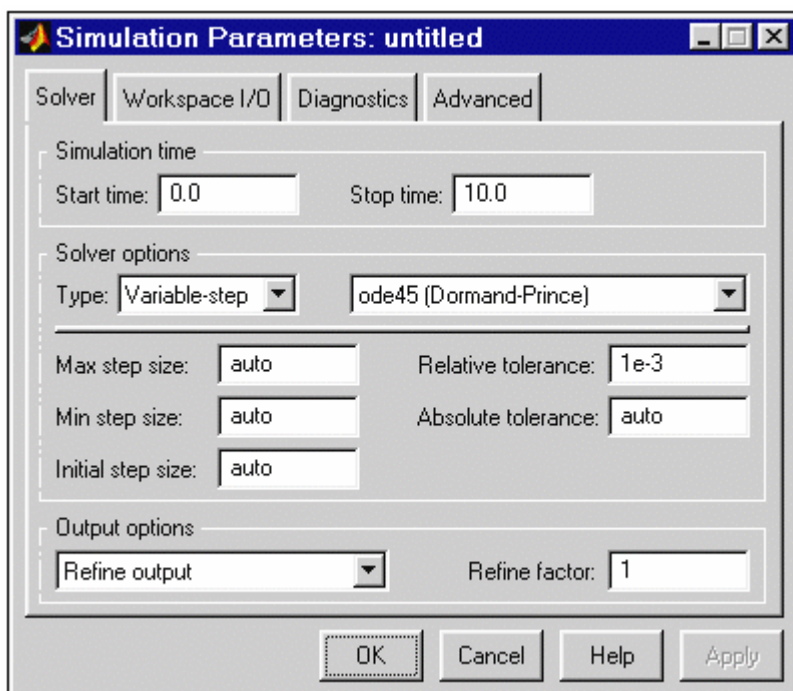


Рисунок 4.1 – Панель управления

Окно настройки параметров расчета имеет 4 вкладки:

- **Solver (Расчет)** – Установка параметров расчета модели.
- **Workspace I/O (Ввод/вывод данных в рабочую область)** – Установка параметров обмена данными с рабочей областью **MATLAB**.
- **Diagnostics (Диагностика)** – Выбор параметров диагностического режима.
- **Advanced (Дополнительно)** – Установка дополнительных параметров.

Установка параметров расчета модели выполняется с помощью элементов управления, размещенных на вкладке **Solver**. Эти элементы разделены на три группы (рисунок 4.1):

- **Simulation time** (Интервал моделирования, или, иными словами, время

расчета),

- **Solver options** (Параметры расчета),
- **Output options** (Параметры вывода).

4.1. Установка параметров расчета модели

4.1.1. Simulation time (Интервал моделирования или время расчета)

Время расчета задается указанием начального (**Start time**) и конечного (**Stop time**) значений времени расчета. Начальное время, как правило, задается равным нулю. Величина конечного времени задается пользователем исходя из условий решаемой задачи.

4.1.2. Solver options (Параметры расчета)

При выборе параметров расчета необходимо указать способ моделирования (**Type**) и метод расчета нового состояния системы. Для параметра **Type** доступны два варианта – с фиксированным (**Fixed-step**) или с переменным (**Variable-step**) шагом. Как правило, **Variable-step** используется для моделирования непрерывных систем, а **Fixed-step** – для дискретных.

Список методов расчета нового состояния системы содержит несколько вариантов. Первый вариант (**discrete**) используется для расчета дискретных систем. Остальные методы используются для расчета непрерывных систем. Эти методы различны для переменного (**Variable-step**) и для фиксированного (**Fixed-step**) шага времени, но, по сути, представляют собой процедуры решения систем дифференциальных уравнений. Подробное описание каждого из методов расчета состояний системы приведено во встроенной справочной системе **MATLAB**.

Ниже двух раскрывающихся списков **Type** находится область, содержимое которой меняется в зависимости от выбранного способа изменения модельного времени. При выборе **Fixed-step** в данной области

появляется текстовое поле **Fixed-step size** (величина фиксированного шага), позволяющее указывать величину шага моделирования (см. рисунок 4.2).

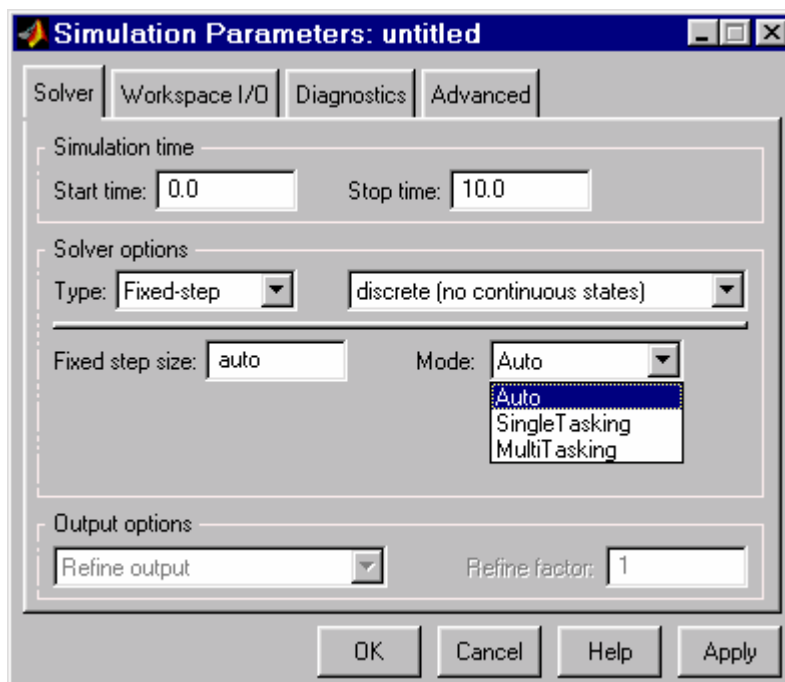


Рисунок 4.2 – Вкладка **Solver** при выборе фиксированного шага расчета

Величина шага моделирования по умолчанию устанавливается системой автоматически (**auto**). Требуемая величина шага может быть введена вместо значения **auto** либо в форме числа, либо в виде вычисляемого выражения (то же самое относится и ко всем параметрам устанавливаемым системой автоматически).

При выборе **Fixed-step** необходимо также задать режим расчета (**Mode**).

Для параметра **Mode** доступны три варианта:

- **MultiTasking** (Многозадачный) – необходимо использовать, если в модели присутствуют параллельно работающие подсистемы, и результат работы модели зависит от временных параметров этих подсистем. Режим позволяет выявить несоответствие скорости и дискретности сигналов, пересылаемых блоками друг другу.
- **SingleTasking** (Однозадачный) – используется для тех моделей, в которых недостаточно строгая синхронизация работы отдельных составляющих не влияет на конечный результат моделирования.

- **Auto** (Автоматический выбор режима) – позволяет **Simulink** автоматически устанавливать режим **MultiTasking** для тех моделей, в которых используются блоки с различными скоростями передачи сигналов и режим **SingleTasking** для моделей, в которых содержатся блоки, оперирующие одинаковыми скоростями.

При выборе **Variable-step** в области появляются поля для установки трех параметров:

- **Max step size** – максимальный шаг расчета. По умолчанию он устанавливается автоматически (**auto**) и его значение в этом случае равно $(\text{SfopTime} - \text{StartTime})/50$. Довольно часто это значение оказывается слишком большим, и наблюдаемые графики представляют собой ломаные (а не плавные) линии. В этом случае величину максимального шага расчета необходимо задавать явным образом.
- **Min step size** – минимальный шаг расчета.
- **Initial step size** – начальное значение шага моделирования.

При моделировании непрерывных систем с использованием переменного шага необходимо указать точность вычислений: относительную (**Relative tolerance**) и абсолютную (**Absolute tolerance**). По умолчанию они равны соответственно 10^{-3} и **auto**.

4.1.3. Output options (Параметры вывода)

В нижней части вкладки **Solver** задаются настройки параметров вывода выходных сигналов моделируемой системы (**Output options**). Для данного параметра возможен выбор одного из трех вариантов:

- **Refine output** (Скорректированный вывод) – позволяет изменять дискретность регистрации модельного времени и тех сигналов, которые сохраняются в рабочей области **MATLAB** с помощью блока **To Workspace**. Установка величины дискретности выполняется в строке редактирования **Refine factor**, расположенной справа. По умолчанию значение **Refine factor** равно 1, это означает, что регистрация производится с шагом $Dt = 1$ (то

есть для каждого значения модельного времени). Если задать **Refine factor** равным 2, будет регистрироваться каждое второе значение сигналов, 3 – каждое третье т. д. Параметр **Refine factor** может принимать только целые положительные значения.

- **Produce additional output** (Дополнительный вывод) – обеспечивает дополнительную регистрацию параметров модели в заданные моменты времени; их значения вводятся в строке редактирования (в этом случае она называется **Output times**) в виде списка, заключенного в квадратные скобки. При использовании этого варианта базовый шаг регистрации (Dt) равен 1. Значения времени в списке **Output times** могут быть дробными числами и иметь любую точность.
- **Produce specified output only** (Формировать только заданный вывод) – устанавливает вывод параметров модели только в заданные моменты времени, которые указываются в поле **Output times** (Моменты времени вывода).

4.2. Установка параметров обмена с рабочей областью

Элементы, позволяющие управлять вводом и выводом в рабочую область **MATLAB** промежуточных данных и результатов моделирования, расположены на вкладке **Workspace I/O** (рисунок 4.3).

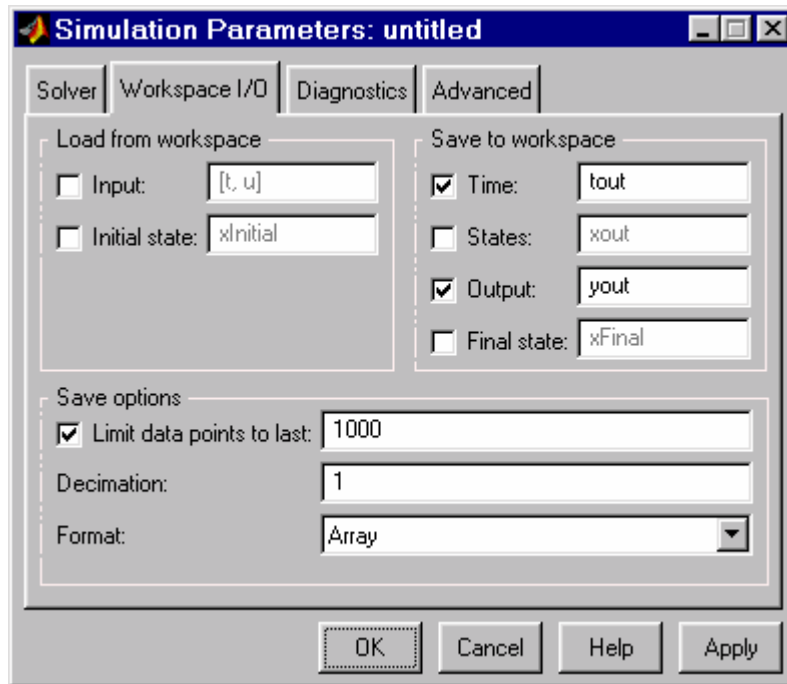


Рисунок 4.3 – Вкладка **Workspace I/O** диалогового окна установки параметров моделирования

Элементы вкладки разделены на 3 поля:

- **Load from workspace** (Загрузить из рабочей области). Если флажок **Input** (Входные данные) установлен, то в расположенном справа текстовом поле можно ввести формат данных, которые будут считываться из рабочей области **MATLAB**. Установка флажка **Initial State** (Начальное состояние) позволяет ввести в связанном с ним текстовом поле имя переменной, содержащей параметры начального состояния модели. Данные, указанные в полях **Input** и **Initial State**, передаются в исполняемую модель посредством одного или более блоков **In** (из раздела библиотеки **Sources**).
- **Save to workspace** (Записать в рабочую область) – Позволяет установить режим вывода значений сигналов в рабочую область **MATLAB** и задать их имена.
- **Save options** (Параметры записи) – Задаёт количество строк при передаче переменных в рабочую область. Если флажок **Limit rows to last** установлен, то в поле ввода можно указать количество передаваемых

строк (отсчет строк производится от момента завершения расчета). Если флажок не установлен, то передаются все данные. Параметр **Decimation** (Исключение) задает шаг записи переменных в рабочую область (аналогично параметру **Refine factor** вкладки **Solver**). Параметр **Format** (формат данных) задает формат передаваемых в рабочую область данных. Доступные форматы **Array** (Массив), **Structure** (Структура), **Structure With Time** (Структура с дополнительным полем – «время»).

4.3. Установка параметров диагностирования модели

Вкладка **Diagnostics** (рисунок 4.4) позволяет изменять перечень диагностических сообщений, выводимых **Simulink** в командном окне **MATLAB**, а также устанавливать дополнительные параметры диагностики модели. Сообщения об ошибках или проблемных ситуациях, обнаруженных **Simulink** в ходе моделирования и требующих вмешательства разработчика, выводятся в командном окне **MATLAB**. Исходный перечень таких ситуаций и вид реакции на них приведен в списке на вкладке **Diagnostics**. Разработчик может указать вид реакции на каждое из них, используя группу переключателей в поле **Action** (они становятся доступны, если в списке выбрано одно из событий):

- **None** – игнорировать,
- **Warning** – выдать предупреждение и продолжить моделирование,
- **Error** – выдать сообщение об ошибке и остановить сеанс моделирования.

Выбранный вид реакции отображается в списке рядом с наименованием события.

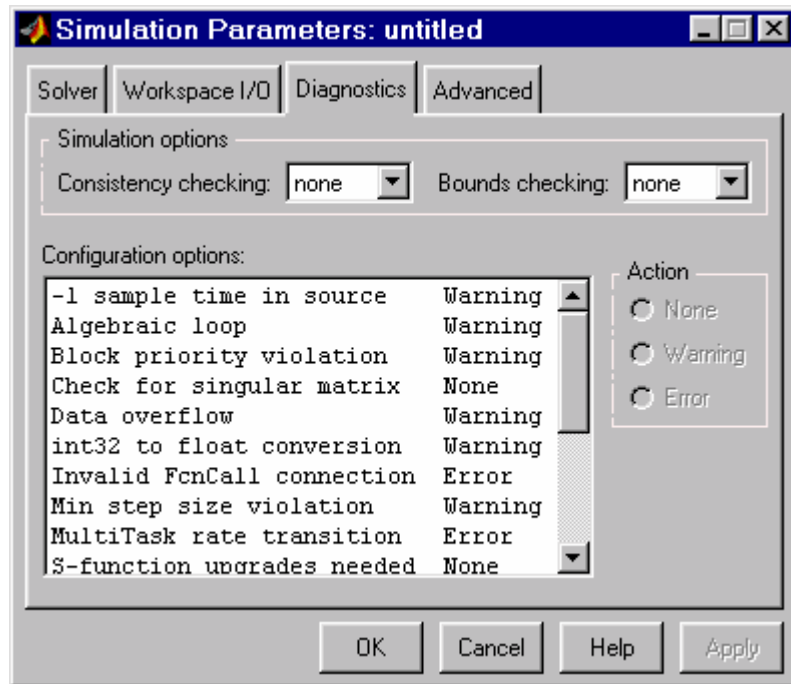


Рисунок 4.4 – Вкладка **Diagnostics** окна установки параметров моделирования

4.4. Выполнение расчета

Запуск расчета выполняется с помощью выбора пункта меню **Simulation/Start**. Процесс расчета можно завершить досрочно, выбрав пункт меню **Simulation/Stop**. Расчет также можно остановить (**Simulation/Pause**) и затем продолжить (**Simulation/Continue**).

5. Обзор библиотеки блоков Simulink

5.1. Sources - источники сигналов

5.1.1. Источник постоянного сигнала Constant

Назначение: Задаёт постоянный по уровню сигнал.

Параметры:

1. **Constant value** – Постоянная величина.
2. **Interpret vector parameters as 1-D** Интерпретировать вектор параметров как одномерный (при установленном флажке).

Данный параметр встречается у большинства блоков библиотеки **Simulink**. В дальнейшем он рассматриваться не будет.

Значение константы может быть действительным или комплексным числом, вычисляемым выражением, вектором или матрицей.

Рисунок 5.1 иллюстрирует применение этого источника и измерение его выходного сигнала с помощью цифрового индикатора **Display**.

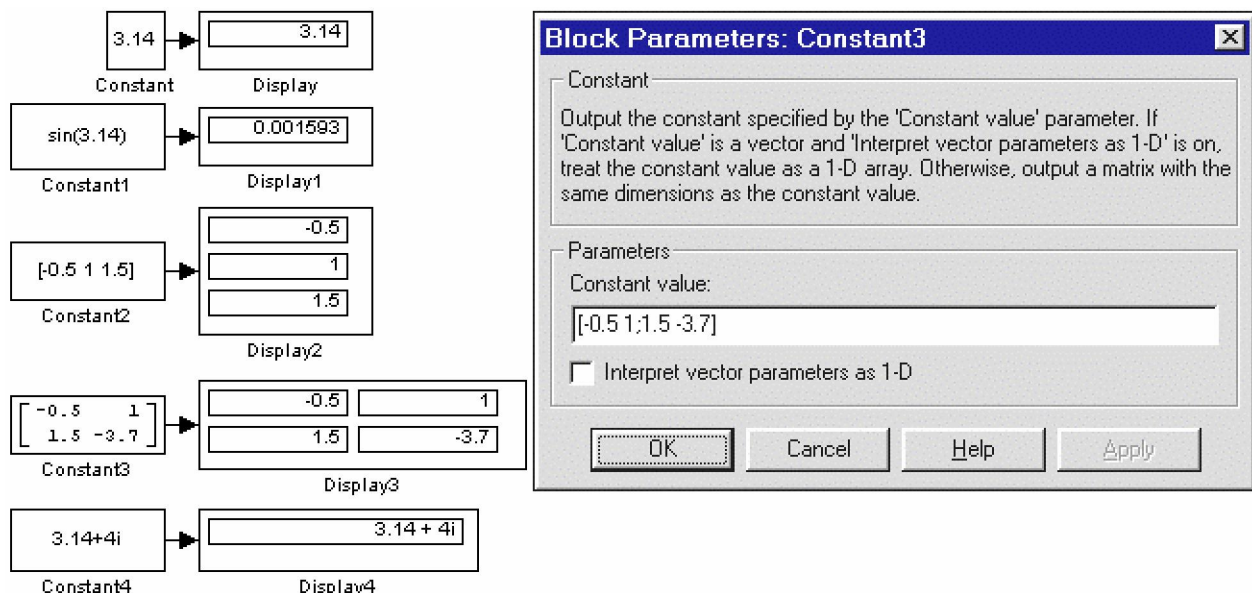


Рисунок 5.1 – Источник постоянного воздействия **Constant**

5.1.2. Генератор ступенчатого сигнала Step

Назначение: Формирует ступенчатый сигнал.

Параметры

1. **Step time** – Время наступления перепада сигнала (с).
2. **Initial value** – Начальное значение сигнала.
3. **Final value** – Конечное значение сигнала.

Перепад может быть как в большую сторону (конечное значение больше чем начальное), так и в меньшую (конечное значение меньше чем начальное). Значения начального и конечного уровней могут быть не только положительными, но и отрицательными (например, изменение сигнала с уровня -5 до уровня -3).

На рисунке 5.2 показано использование генератора ступенчатого сигнала.

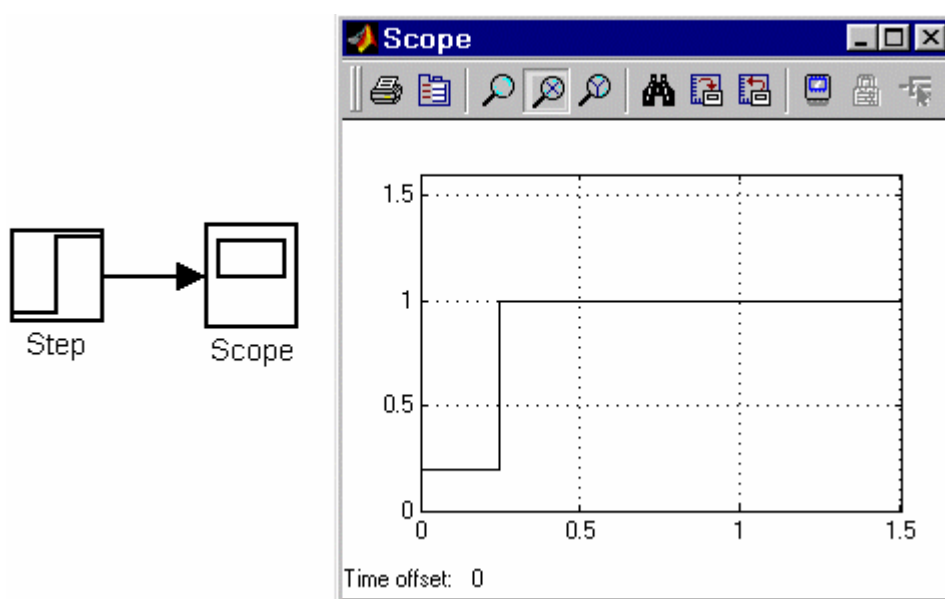


Рисунок 5.2 – Блок Step

5.1.3. Генератор сигналов Signal Generator

Назначение: Формирует один из четырех видов периодических сигналов:

1. **sine** – Синусоидальный сигнал.
2. **square** – Прямоугольный сигнал.

3. **sawtooth** – Пилообразный сигнал.

4. **random** – Случайный сигнал.

Параметры:

1. **Wave form** – Вид сигнала.

2. **Amplitude** – Амплитуда сигнала.

3. **Frequency** – Частота (рад/с).

4. **Units** – Единицы измерения частоты. Может принимать два значения:

- **Hertz** – Гц.
- **rad/sec** – рад/с.

На рисунок 5.3 показано применение этого источника при моделировании прямоугольного сигнала.

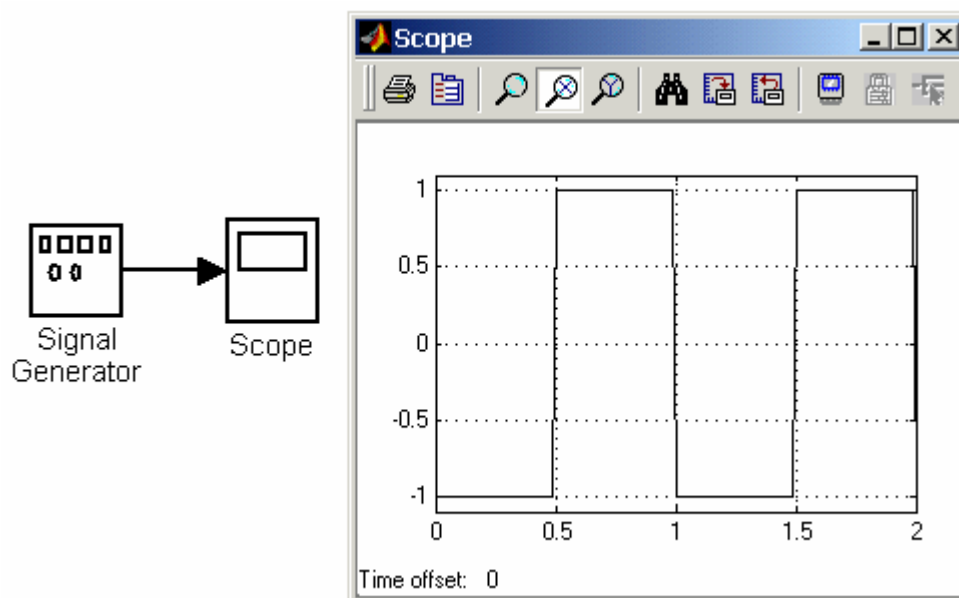


Рисунок 5.3 – Блок генератора сигналов

5.1.4. Источник случайного сигнала с равномерным распределением

Uniform Random Number

Назначение: Формирование случайного сигнала с равномерным распределением.

Параметры:

1. **Minimum** – Минимальный уровень сигнала.

2. **Maximum** – Максимальный уровень сигнала.

3. **Initial seed** – Начальное значение.

Пример использования блока и график его выходного сигнала представлен на рисунок 5.4

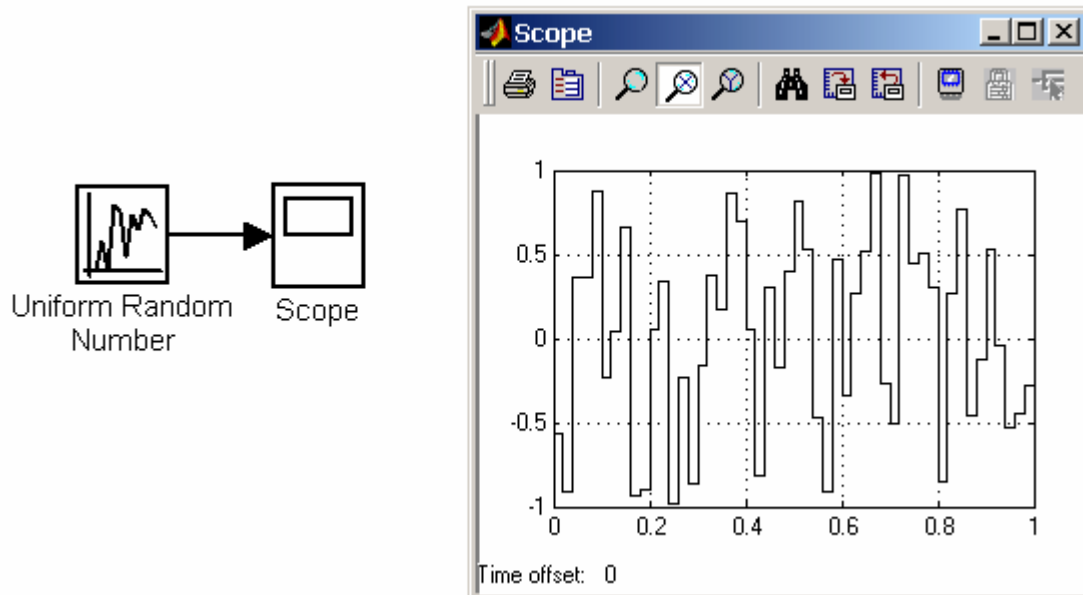


Рисунок 5.4 – Источник случайного сигнала с равномерным распределением

5.1.5. Источник случайного сигнала с нормальным распределением

Random Number

Назначение: Формирование случайного сигнала с нормальным распределением уровня сигнала.

Параметры:

1. **Mean** – Среднее значение сигнала
2. **Variance** – Дисперсия (среднеквадратическое отклонение).
3. **Initial seed** – Начальное значение.

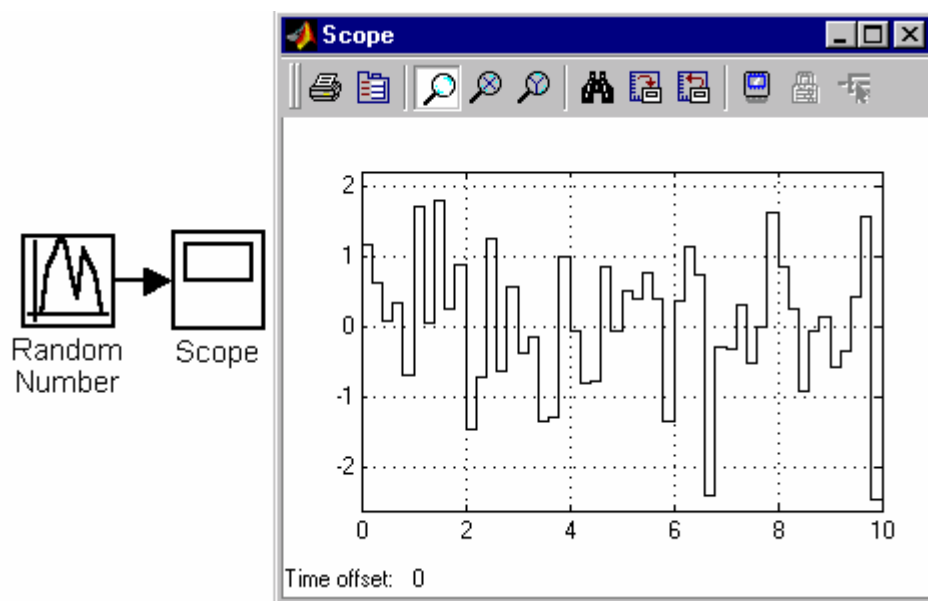


Рисунок 5.5 – Источник случайного сигнала с нормальным распределением

5.1.6. Источник импульсного сигнала Pulse Generator

Назначение: Формирование прямоугольных импульсов.

Параметры:

1. **Pulse Type** – Способ формирования сигнала. Может принимать два значения:
 - **Time-based** – По текущему времени.
 - **Sample-based** – По величине модельного времени и количеству расчетных шагов.
2. **Amplitude** – Амплитуда.
3. **Period** – Период. Задается в секундах для **Time-based Pulse Type** или в шагах модельного времени для **Sample-based Pulse Type**.
4. **Pulse width** – Ширина импульсов. Задается в процентах по отношению к периоду для **Time-based Pulse Type** или в шагах модельного времени для **Sample-based Pulse Type**.
5. **Phase delay** – Фазовая задержка. Задается в секундах для **Time-based Pulse Type** или в шагах модельного времени для **Sample-based Pulse Type**.
6. **Sample time** – Шаг модельного времени. Задается для **Sample-based Pulse Type**.

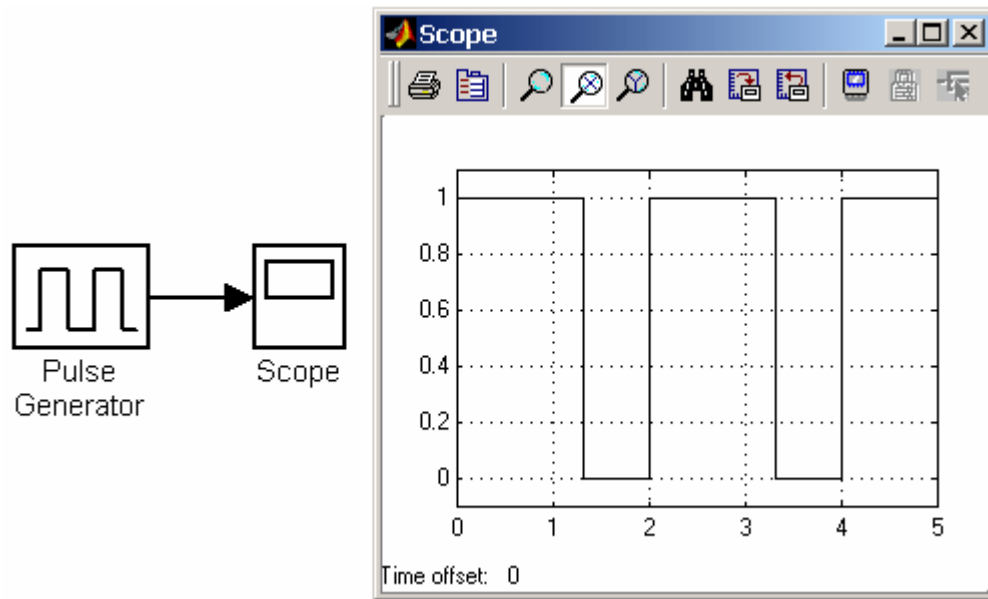


Рисунок 5.6 – Источник прямоугольных импульсов

5.1.7. Источник временного сигнала Clock

Назначение: Формирует сигнал, величина которого на каждом шаге расчета равна текущему времени моделирования.

Параметры:

1. **Decimation** – Шаг, с которым обновляются показания времени на изображении источника (в том случае, если установлен флажок параметра **Display time**). Параметр задается как количество шагов расчета. Например, если шаг расчета модели в окне диалога **Simulation parameters** установлен равным **0,01** с, а параметр **Decimation** блока **Clock** задан равным **1000**, то обновление показаний времени будет производиться каждые **10** с модельного времени.

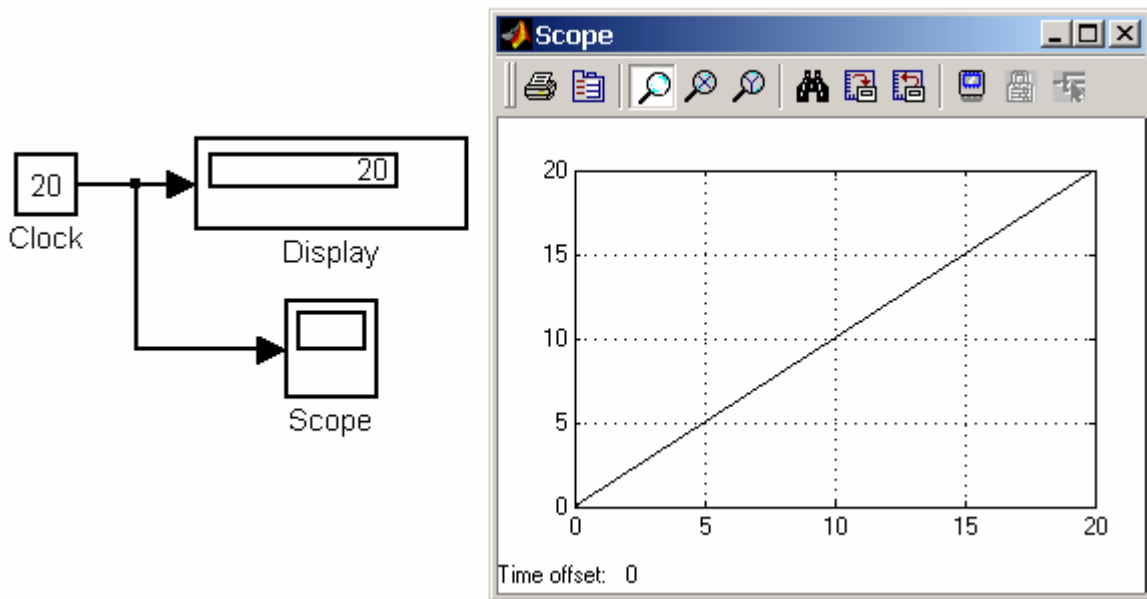


Рисунок 5.7 – Источник временного сигнала

2. **Display time** – Отображение значения времени в блоке источника.

На рисунке 5.7 показан пример работы данного источника.

5.1.8. Цифровой источник времени Digital Clock

Назначение: Формирует дискретный временной сигнал.

Параметр:

Sample time – Шаг модельного времени (с).

На рисунке 5.8 показана работа источника **Digital Clock**

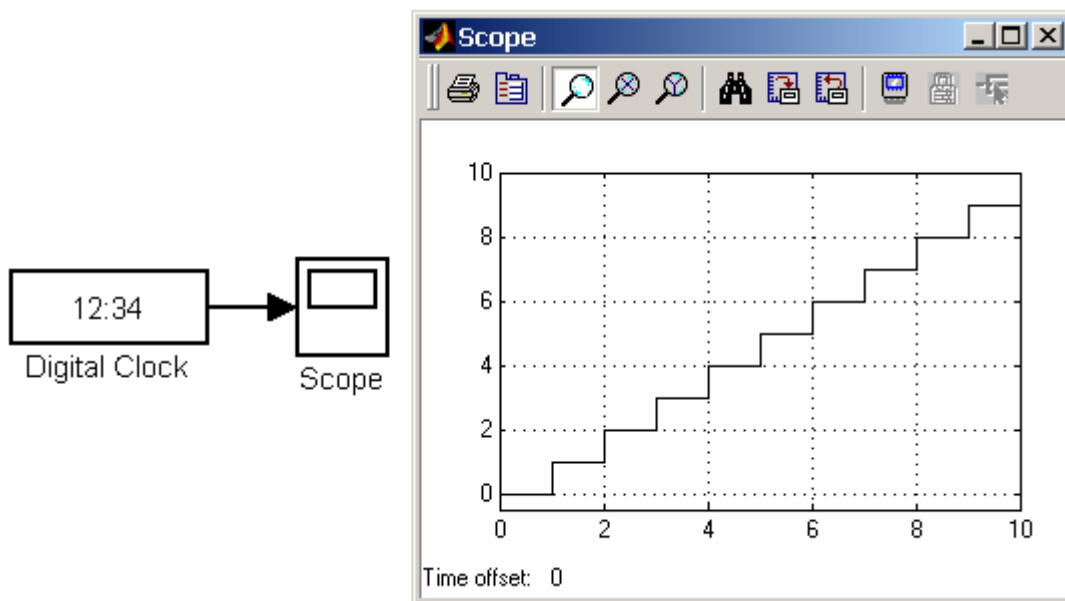


Рисунок 5.8 – Цифровой источник временного сигнала

5.2. Sinks - приемники сигналов

5.2.1. Осциллограф Scope

Назначение:

Строит графики исследуемых сигналов в функции времени. Позволяет наблюдать за изменениями сигналов в процессе моделирования.

Изображение блока и окно для просмотра графиков показаны на рисунке 5.9.

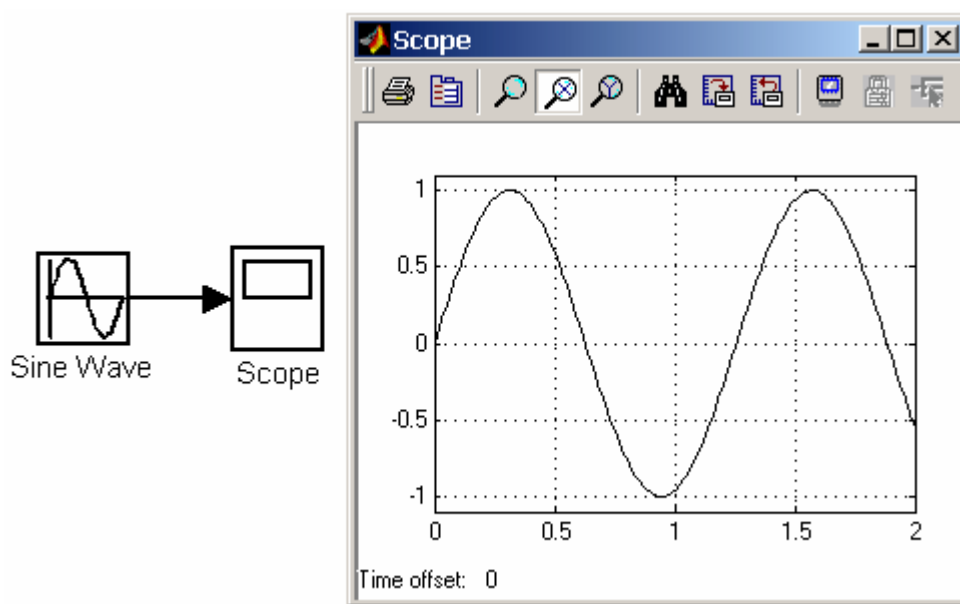


Рисунок 5.9 – Осциллограф **Scope**

Для того чтобы открыть окно просмотра сигналов, необходимо выполнить двойной щелчок левой клавишей мыши на изображении блока. Это можно сделать на любом этапе расчета (как до начала расчета, так и после него, а также во время расчета). В том случае, если на вход блока поступает векторный сигнал, то кривая для каждого элемента вектора строится отдельным цветом.

Настройка окна осциллографа выполняется с помощью панелей инструментов (рисунок 5.10).

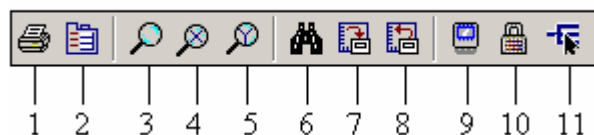








Рисунок 5.10 – Панель инструментов блока **Score**

Панель инструментов содержит 11 кнопок:

1. **Print** – печать содержимого окна осциллографа.
2. **Parameters** – доступ к окну настройки параметров.
3. **Zoom** – увеличение масштаба по обеим осям.
4. **Zoom X-axis** – увеличение масштаба по горизонтальной оси.
5. **Zoom Y-axis** – увеличение масштаба по вертикальной оси.
6. **Autoscale** – автоматическая установка масштабов по обеим осям.
7. **Save current axes settings** – сохранение текущих настроек окна.
8. **Restore saved axes settings** – установка ранее сохраненных настроек окна.
9. **Floating scope** – перевод осциллографа в «свободный» режим.
10. **Lock/Unlock axes selection** – закрепить/разорвать связь между текущей координатной системой окна и отображаемым сигналом. Инструмент доступен, если включен режим **Floating scope**.
11. **Signal selection** – выбор сигналов для отображения. Инструмент доступен, если включен режим **Floating scope**.

Изменение масштабов отображаемых графиков можно выполнять несколькими способами:

1. Нажать соответствующую кнопку (,  или ) и щелкнуть один раз левой клавишей мыши в нужном месте графика. Произойдет 2,5 кратное увеличение масштаба.
2. Нажать соответствующую кнопку (,  или ) и, нажав левую клавишу мыши, с помощью динамической рамки или отрезка указать область графика для увеличенного изображения. Рисунок 5.11 поясняет этот процесс.

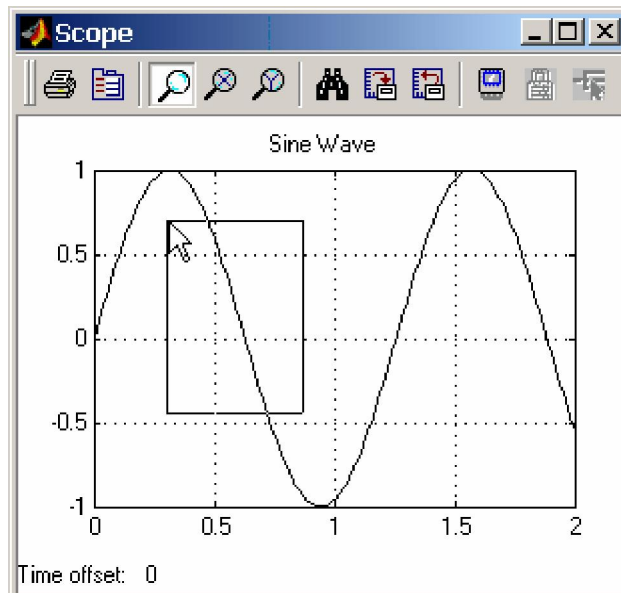


Рисунок 5.11 – Увеличение масштаба графика

- Щелкнуть правой клавишей мыши в окне графиков и выбрать команду **Axes properties...** в контекстном меню. Откроется окно свойств графика, в котором с помощью параметров **Y-min** и **Y-max** можно указать предельные значения вертикальной оси. В этом же окне можно указать заголовок графика (**Title**), заменив выражение **%<SignalLabel>** в строке ввода. Окно свойств показано на рисунке 5.13.

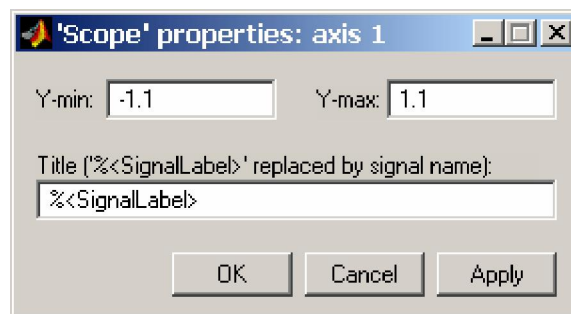



Рисунок 5.13 - Окно свойств графика

Параметры:

Параметры блока устанавливаются в окне **'Scope' parameters**, которое открывается с помощью инструмента  (**Parameters**) панели инструментов.

Окно параметров имеет две вкладки:

- **General** – общие параметры.

• **Data history** – параметры сохранения сигналов в рабочей области MATLAB.

Вкладка общих параметров показана на рисунке 5.14.

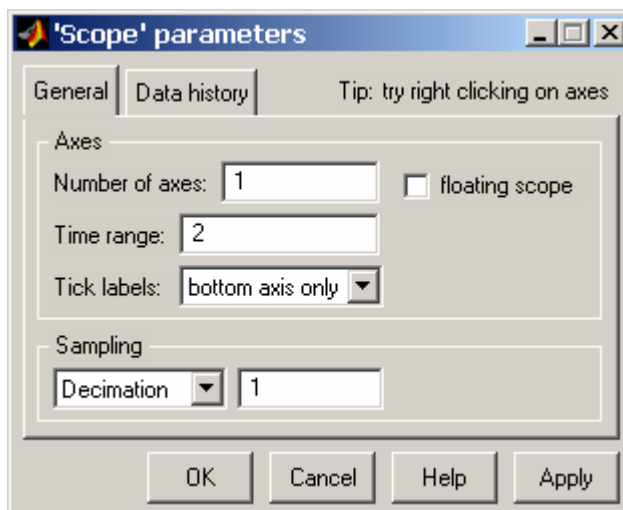


Рисунок 5.14 – Вкладка общих параметров **General**

На вкладке **General** задаются следующие параметры:

1. **Number of axes** – число входов (систем координат) осциллографа. При изменении этого параметра на изображении блока появляются дополнительные входные порты.
2. **Time range** – величина временного интервала, для которого отображаются графики. Если время расчета модели превышает заданное параметром **Time range**, то вывод графика производится порциями, при этом интервал отображения каждой порции графика равен заданному значению **Time range**.
3. **Tick labels** – вывод/скрытие осей и меток осей. Может принимать три значения (выбираются из списка):
 - **all** – подписи для всех осей,
 - **none** – отсутствие всех осей и подписей к ним,
 - **bottom axis only** – подписи горизонтальной оси только для нижнего графика.
4. **Sampling** — установка параметров вывода графиков в окне. Задаёт режим вывода расчетных точек на экран. При выборе **Decimation** кратность

вывода устанавливается числом, задающим шаг выводимых расчетных точек. На рисунке 5.15 и 5.16 показаны графики синусоидальных сигналов рассчитанных с фиксированным шагом **0.1** с. На рисунке 5.15 в окне блока **Scope** выводится каждая расчетная точка (параметр **Decimation** равен 1). На рисунке 5.16 показан вывод каждого второго значения (параметр **Decimation** равен 2). Маркерами на графиках отмечены расчетные точки.

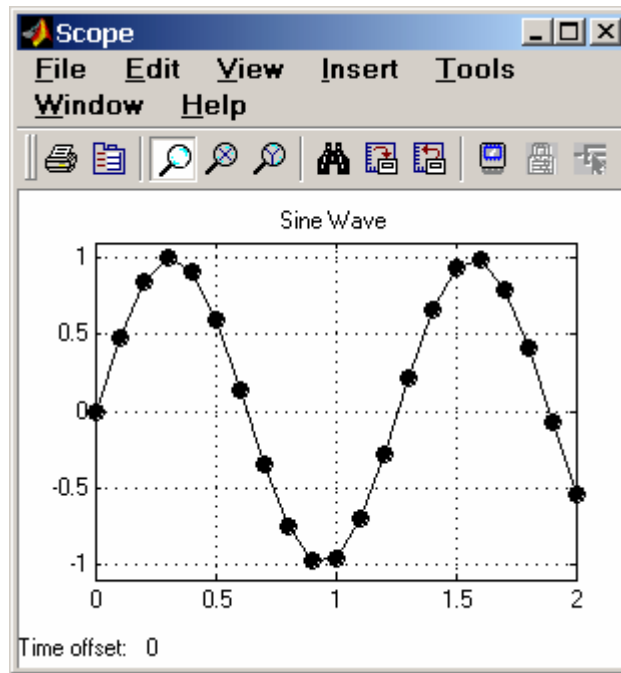


Рисунок 5.15 – Отображение синусоидального сигнала (**Decimation = 1**)

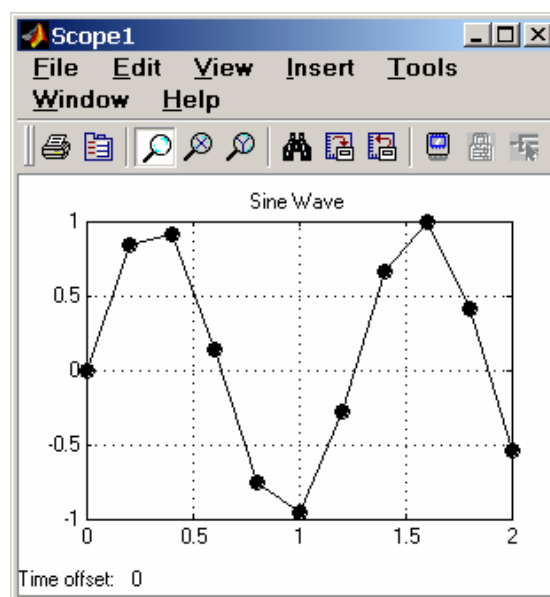


Рисунок 5.16 - Отображение синусоидального сигнала (**Decimation = 2**)

В том случае, если режим вывода расчетных точек задается как **Sample time**, то его числовое значение определяет интервал квантования при отображении сигнала. На рисунок 5.17 показан график синусоидального сигнала, для случая, когда значение параметра **Sample time** равно **0,1**.

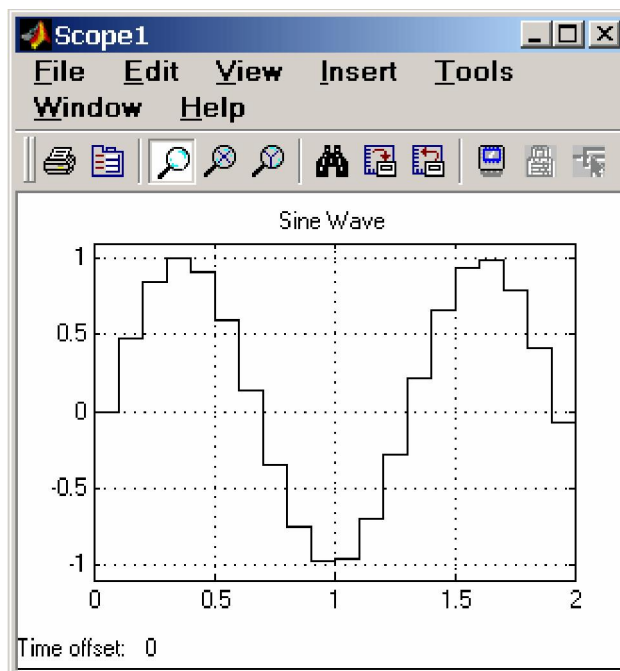


Рисунок 5.17 – Отображение синусоидального сигнала (**Sample time = 0,1**)

5. **floating scope** – перевод осциллографа в «свободный» режим (при установленном флажке).

На вкладке **Data history** (рисунок 5.18) задаются следующие параметры:

Limit data points to last – максимальное количество отображаемых расчетных точек графика. При превышении этого числа начальная часть графика обрезается. В том случае, если флажок параметра **Limit data points to last** не установлен, то **Simulink** автоматически увеличит значение этого параметра для отображения всех расчетных точек.

Save data to workspace – сохранение значений сигналов в рабочей области **MATLAB**.

Variable name – имя переменной для сохранения сигналов в рабочей области **MATLAB**.

Format – формат данных при сохранении в рабочей области **MATLAB**.

Может принимать значения:

- **Array** – массив,
- **Structure** – структура,
- **Structure with time** – структура с дополнительным полем «время».

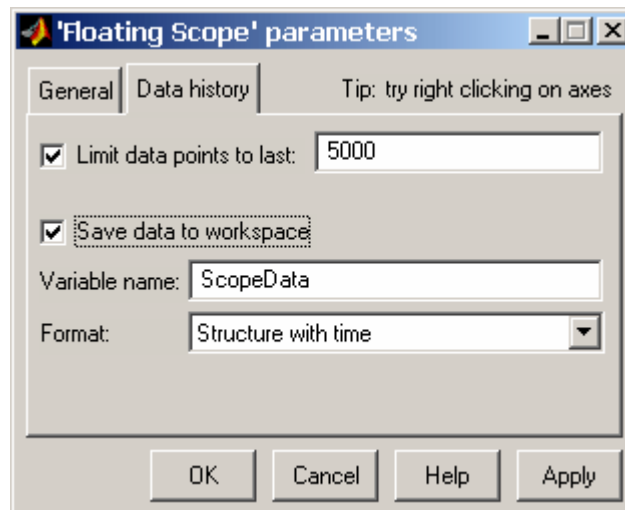


Рисунок 5.18 – Вкладка **Data history**

5.2.2. Графопостроитель XY Graph

Назначение:

Строит график одного сигнала в функции другого (график вида $Y(X)$).

Параметры:

x-min – Минимальное значение сигнала по оси **X**.

x-max – Максимальное значение сигнала по оси **X**

y-min – Минимальное значение сигнала по оси **Y**. **y-max** – Максимальное значение сигнала по оси **Y** **Sample time** – шаг модельного времени.

Блок имеет два входа. Верхний вход предназначен для подачи сигнала, который является аргументом (**X**), нижний – для подачи значений функции (**Y**).

На рисунке 5.19, в качестве примера использования графопостроителя, показано построение фазовой траектории колебательного

звена.

Графопостроитель можно использовать и для построения временных зависимостей. Для этого на первый вход следует подать временной сигнал с выхода блока **Clock**. Пример такого использования графопостроителя показан на рисунке 5.20.

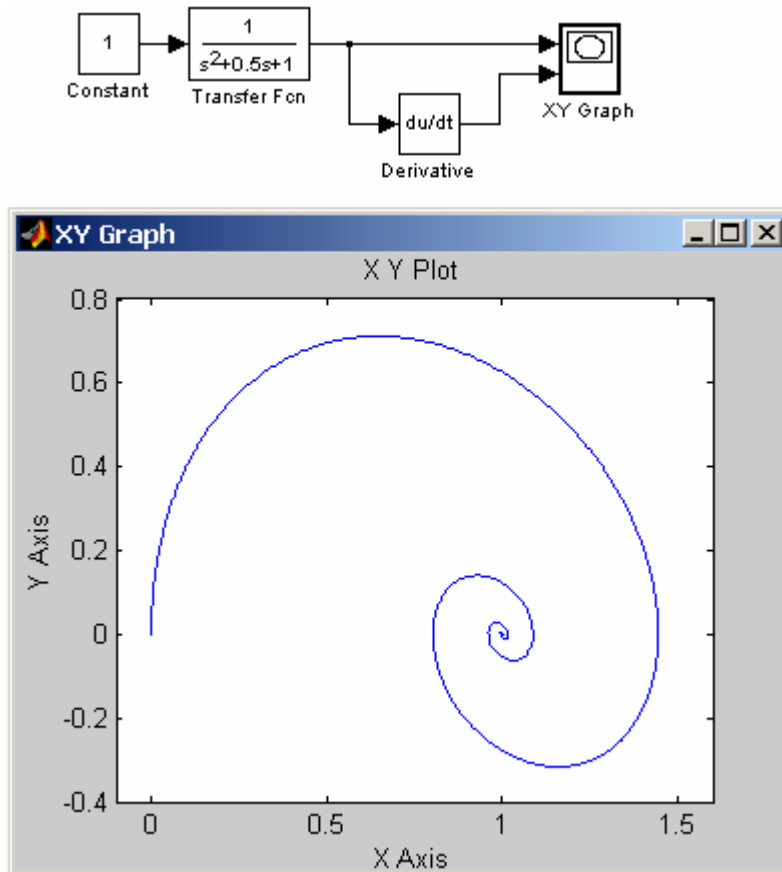


Рисунок 5.20 – Пример использования графопостроителя XY Graph

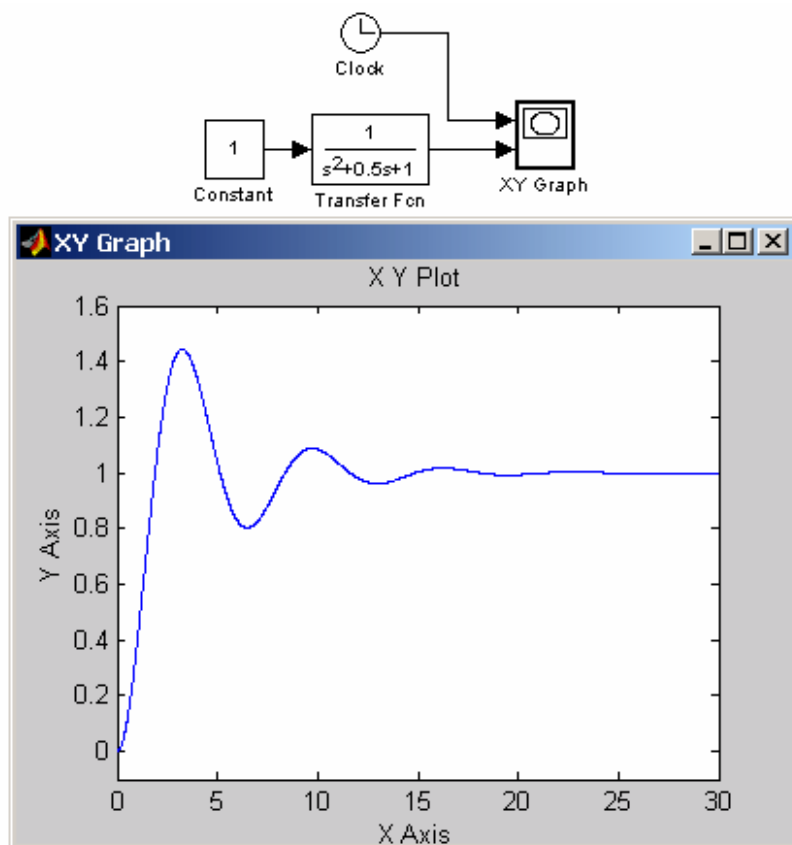


Рисунок 5.21 – Блок XY Graph для отображения временных зависимостей

5.2.3. Цифровой дисплей Display

Назначение: Отображает значение сигнала в виде числа.

Параметры:

1. **Format** – формат отображения данных. Параметр **Format** может принимать следующие значения:
 - **short** – 5 значащих десятичных цифр.
 - **long** – 15 значащих десятичных цифр.
 - **short_e** – 5 значащих десятичных цифр и 3 символа степени десяти.
 - **long_e** – 15 значащих десятичных цифр и 3 символа степени десяти.
 - **bank** – «денежный» формат. Формат с фиксированной точкой и двумя десятичными цифрами в дробной части числа.
2. **Decimation** – кратность отображения входного сигнала. При **Decimation = 1** отображается каждое значение входного сигнала, при **Decimation = 2** – каждое второе значение, при **Decimation = 3** – каждое

третье значение и т.д.

3. Sample time – шаг модельного времени. Определяет дискретность отображения данных.

4. Floating display (флажок)– перевод блока в «свободный» режим. В данном режиме входной порт блока отсутствует, а выбор сигнала для отображения выполняется щелчком левой клавиши мыши на соответствующей линии связи. В этом режиме для параметра расчета **Signal storage reuse** должно быть установлено значение **off** (вкладка **Advanced** в окне диалога **Simulation parameters...**). На рисунке 5.22 показано применение блока **Display** с использованием различных вариантов параметра **Format**.

Блок **Display** может использоваться для отображения не только скалярных сигналов, но также векторных, матричных и комплексных. Рисунок 5.23 иллюстрирует это. Если все отображаемые значения не могут поместиться в окне блока, в правом нижнем углу блока появляется символ, указывающий на необходимость увеличить размеры блока (см. блок **Display4** на рисунке 5.23).

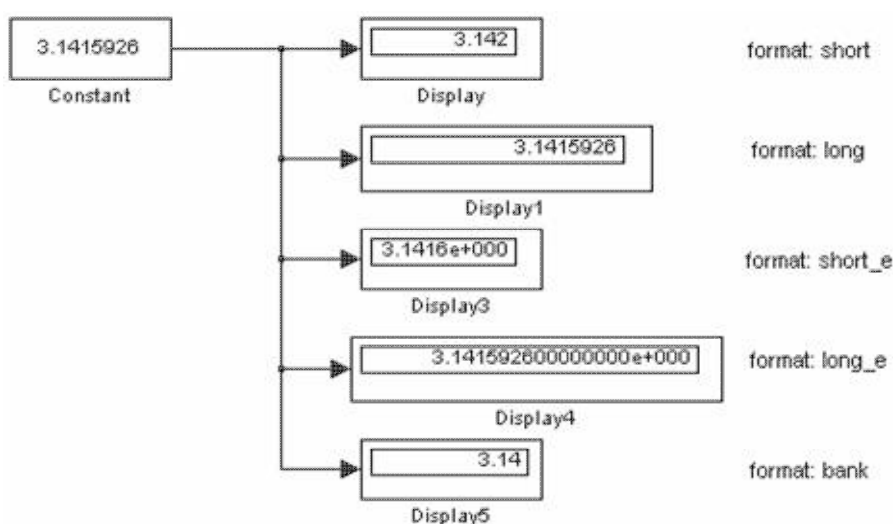


Рисунок 5.22 – Применение блока **Display** с использованием различных вариантов параметра **Format**

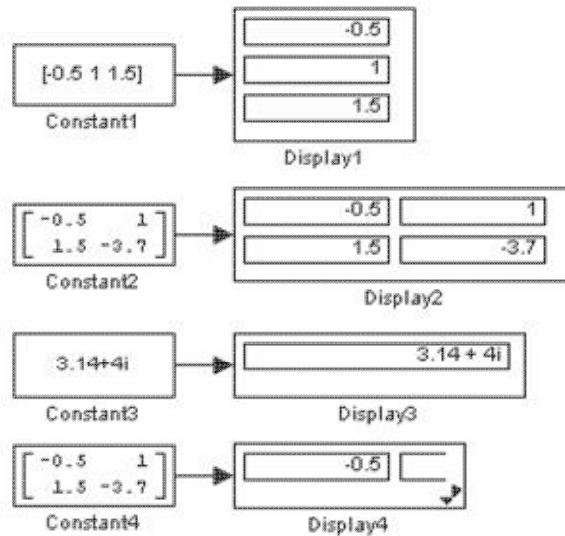


Рисунок 5.23 – Применение блока **Display** для отображения векторных, матричных и комплексных сигналов

5.4. Блок остановки моделирования **Stop Simulation**

Назначение: Обеспечивает завершение расчета, если входной сигнал блока становится не равным нулю.

Параметры: Нет.

При подаче на вход блока ненулевого сигнала **Simulink** выполняет текущий шаг расчета, а затем останавливает моделирование. Если на вход блока подан векторный сигнал, то для остановки расчета достаточно, чтобы один элемент вектора стал ненулевым. На рисунке 5.24 показан пример использования данного блока. В примере остановка расчета происходит, если выходной сигнал блока **Transfer Function** становится большим или равным **0,99**.

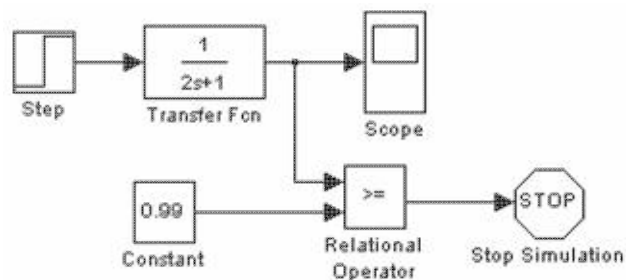


Рисунок 5.24 – Применение блока **Stop Simulation**

5.3. Continuous – аналоговые блоки

5.3.1. Блок Memory

Назначение:

Выполняет задержку входного сигнала на один временной такт.

Параметры:

- **Initial condition** – начальное значение выходного сигнала.
- **Inherit sample time** (флажок) – Наследовать шаг модельного времени.

Если этот флажок установлен, то блок **Memory** использует шаг модельного времени (**Sample time**) такой же, как и в предшествующем блоке.

На рисунок 5.25 показан пример использования блока **Memory** для задержки дискретного сигнала на один временной такт.

5.3.2. Блок фиксированной задержки сигнала Transport Delay

Назначение: Обеспечивает задержку входного сигнала на заданное время.

Параметры:

1. **Time Delay** — Время задержки сигнала (не отрицательное значение).
2. **Initial input** — Начальное значение выходного сигнала.

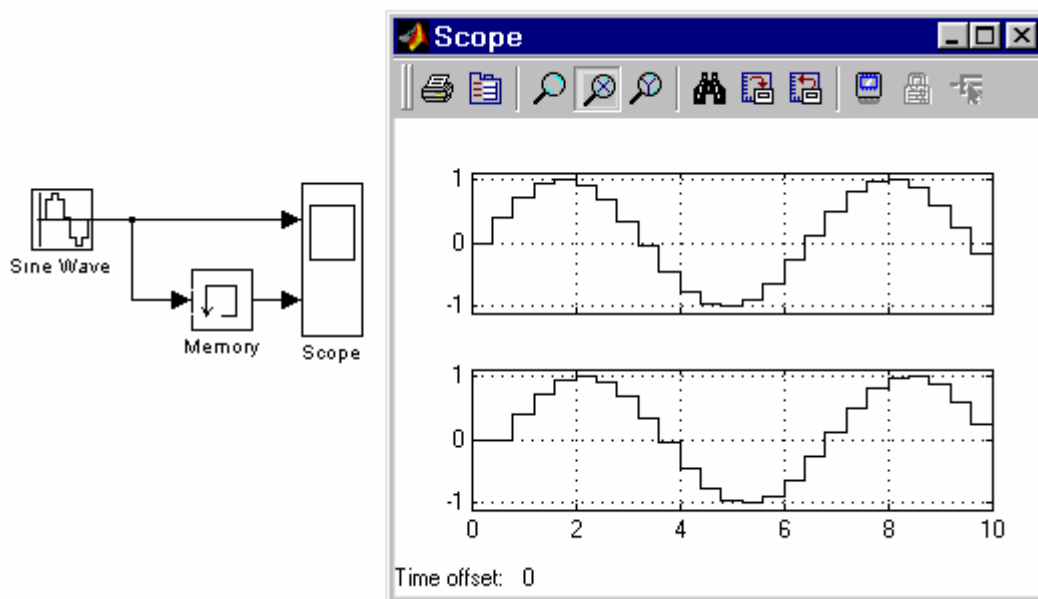


Рисунок 5.25 – Применение блока для задержки сигнала на один временной такт

3. **Buffer size** — Размер памяти, выделяемой для хранения задержанного

сигнала. Задается в байтах числом, кратным восьми (по умолчанию **1024**).

4. Pade order (for linearization) — Порядок ряда Паде, используемого при аппроксимации выходного сигнала. Задается целым положительным числом. При выполнении моделирования, значение сигнала и соответствующее ему модельное время сохраняются во внутреннем буфере блока **Transport Delay**. По истечении времени задержки значение сигнала извлекается из буфера и передается на выход блока. В том случае, если шаги модельного времени не совпадают со значениями моментов времени для записанного в буфер сигнала, блок **Transport Delay** выполняет аппроксимацию выходного сигнала.

В том случае, если начального значения объема памяти буфера не хватит для хранения задержанного сигнала, **Simulink** автоматически выделит дополнительную память. После завершения моделирования в командном окне **MATLAB** появится сообщение с указанием нужного размера буфера.

На рисунке 5.26 показан пример использования блока **Transport Delay** для задержки прямоугольного сигнала на **0,5 с**.

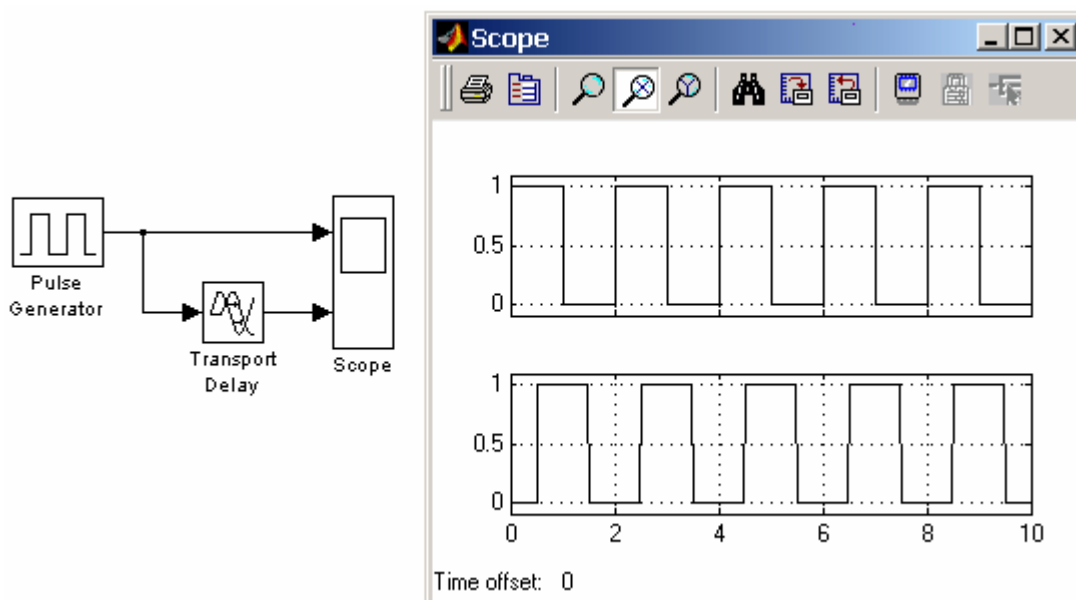


Рисунок 5.26 – Пример использования блока **Transport Delay** для задержки сигнала

5.4. Discrete – дискретные блоки

5.4.1. Блок единичной дискретной задержки Unit Delay

Назначение:

Выполняет задержку входного сигнала на один шаг модельного времени.

Параметры:

1. **Initial condition** – Начальное значение для выходного сигнала.
2. **Sample time** – Шаг модельного времени.

Входной сигнал блока может быть как скалярным, так и векторным. При векторном входном сигнале задержка выполняется для каждого элемента вектора. Блок поддерживает работу с комплексными и действительными сигналами.

На рисунке 5.27 показан пример использования блока для задержки дискретного сигнала на один временной шаг, равный **0,1с**.

5.4.1. Блок дискретного интегратора Discrete-Time Integrator

Назначение:

Блок используется для выполнения операции интегрирования в дискретных системах.

Параметры:

1. **Integration method** – Метод численного интегрирования:

- **Forward Euler** – Прямой метод Эйлера.

Метод использует аппроксимацию $T/(z-1)$ передаточной функции $1/s$.

Выходной сигнал блока рассчитывается по выражению:

$$y(k) = y(k-1) + T * u(k-1),$$

y – выходной сигнал интегратора,

u – входной сигнал интегратора,

T – шаг дискретизации,

k – номер шага моделирования.

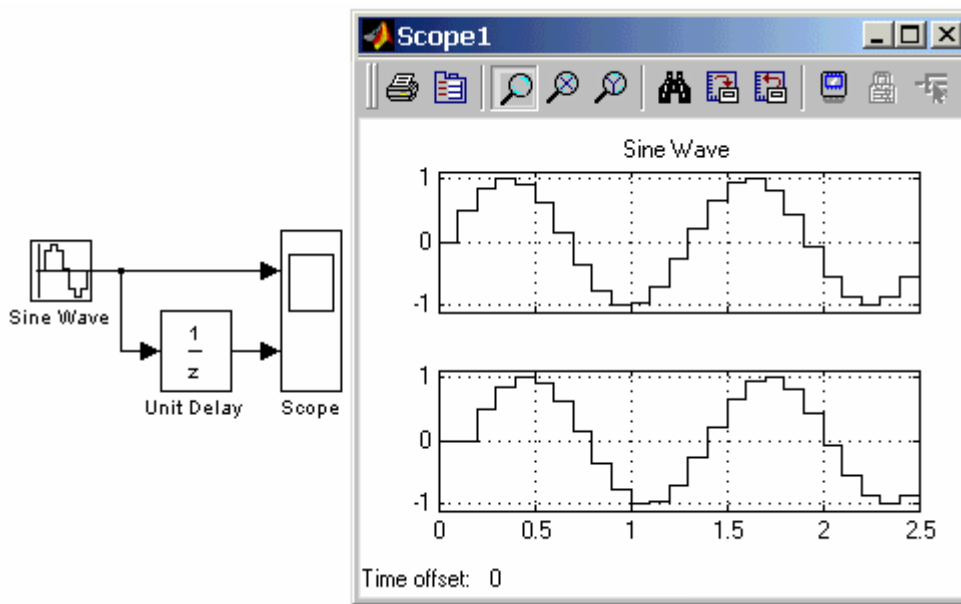


Рисунок 5.27 – Пример использования блока **Unit Delay**

- **Backward Euler** – Обратный метод Эйлера.

Метод использует аппроксимацию $T \times z / (z-1)$ передаточной функции $1/s$.

Выходной сигнал блока рассчитывается по выражению:

$$y(k) = y(k-1) + T \times u(k).$$

- **Trapezoidal** – Метод трапеций.

Метод использует аппроксимацию $T/2 \times (z+1)/(z-1)$ передаточной функции $1/s$.

Выходной сигнал блока рассчитывается по выражению:

$$x(k) = y(k-1) + T/2 \times u(k-1).$$

2. **Sample time** – Шаг дискретизации по времени.

Остальные параметры дискретного интегратора те же, что и у блока аналогового интегратора **Integrator** (библиотека **Continuous**).

На рисунке 5.28 показан пример демонстрирующий все три способа численного интегрирования блока **Discrete-Time Integrator**. Как видно из рисунка, изображение блока меняется в зависимости от выбранного метода интегрирования.

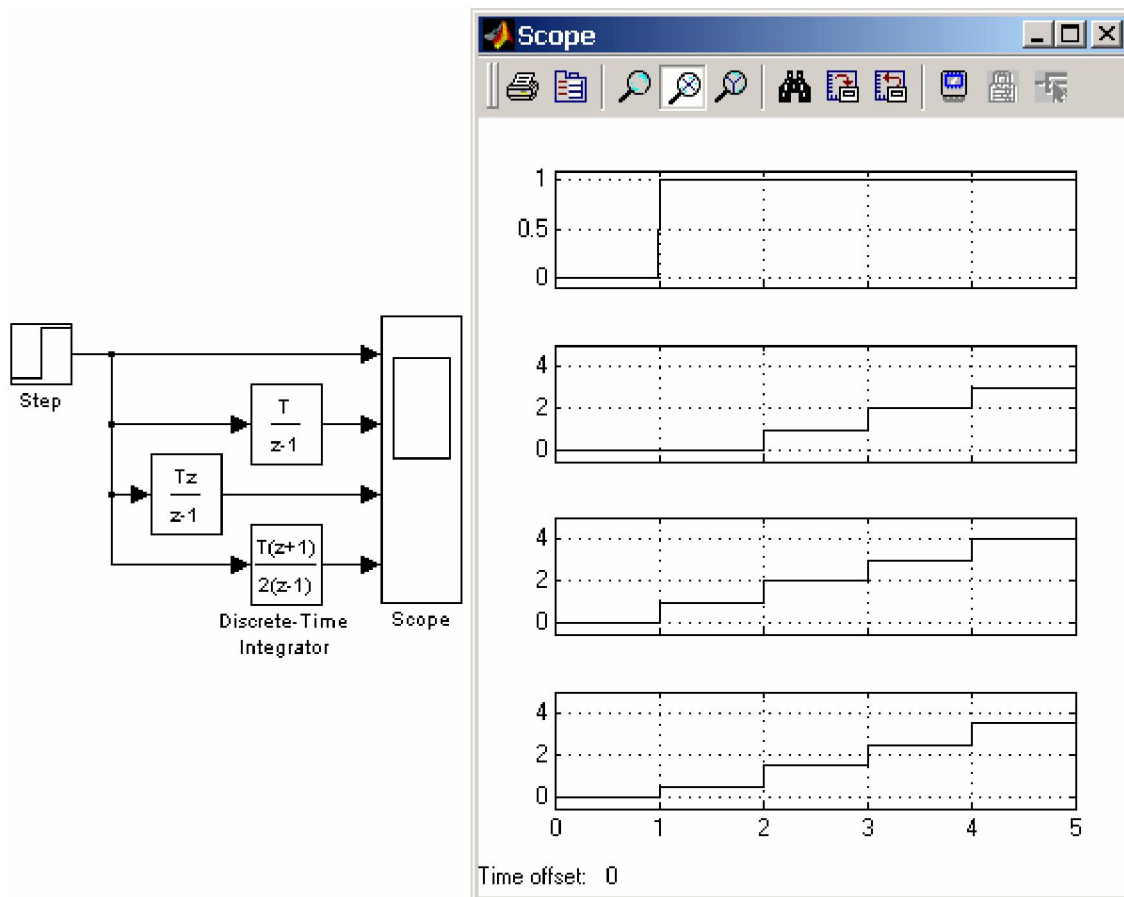


Рисунок 5.28 – Выполнение интегрирования блоками **Discrete-Time Integrator**, реализующими разные численные методы

5.5. Nonlinear - нелинейные блоки

5.5.1. Блок переключателя Switch

Назначение:

Выполняет переключение входных сигналов по сигналу управления.

Параметры:

Threshold – Порог управляющего сигнала.

Блок работает следующим образом: если сигнал управления, подаваемый на средний вход, меньше, чем величина порогового значения **Threshold**, то на выход блока проходит сигнал с первого (верхнего) входа. Если сигнал управления превысит пороговое значение, то на выход блока будет поступать сигнал со второго (нижнего) входа.

На рисунке 5.29 показан пример работы блока **Switch**. В том случае,

когда сигнал на управляющем входе ключа равен 1, на выход блока проходит гармонический сигнал, если же управляющий сигнал равен нулю, то на выход проходит сигнал нулевого уровня от блока **Ground**. Пороговое значение управляющего сигнала задано равным 0,5.

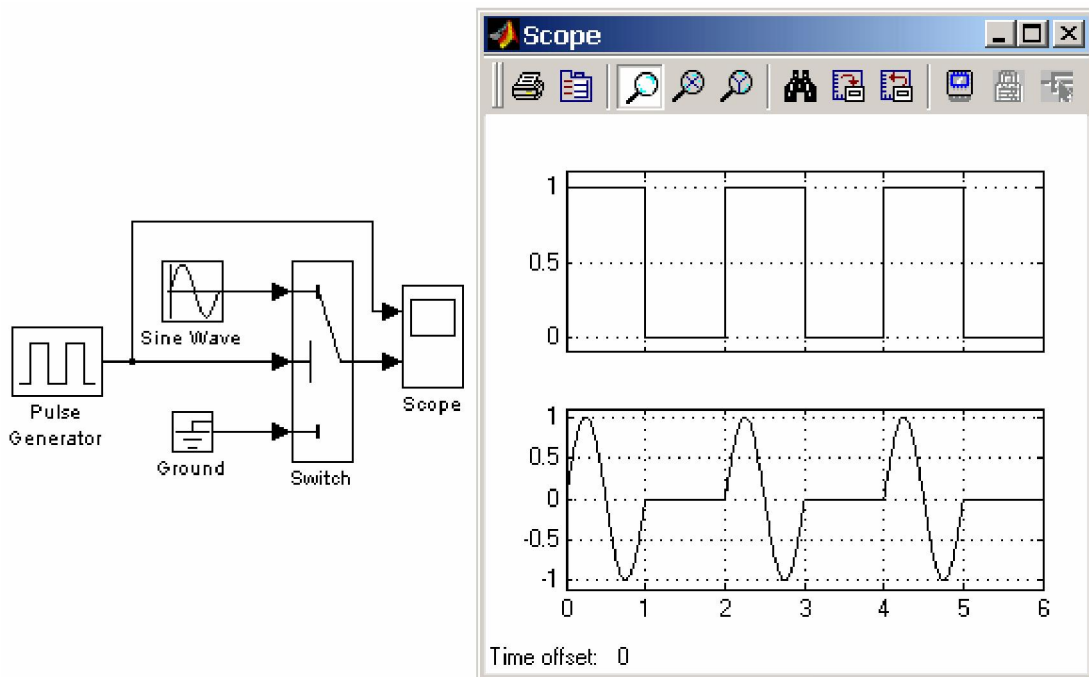


Рисунок 5.29 – Применение переключателя **Switch**

5.5.2. Блок многовходового переключателя **Multiport Switch**

Назначение:

Выполняет переключение входных сигналов по сигналу управления, задающему номер активного входного порта.

Параметры:

Number of inputs – Количество входов.

Блок многовходового переключателя **Multiport Switch**, пропускает на выход сигнал с того входного порта, номер которого равен текущему значению управляющего сигнала. Если управляющий сигнал не является сигналом целого типа, то блок **Multiport Switch** производит отбрасывание дробной части числа, при этом в командном окне **Matlab** появляется предупреждающее сообщение.

На рисунке 5.30 показан пример работы блока **Multiport Switch**.

Управляющий сигнал переключателя имеет три уровня и формируется с помощью блоков **Constant**, **Step**, **Step1** и **Sum**. На выход блока **Multiport Switch**, в зависимости от уровня входного сигнала, проходят гармонические сигналы, имеющие разные частоты.

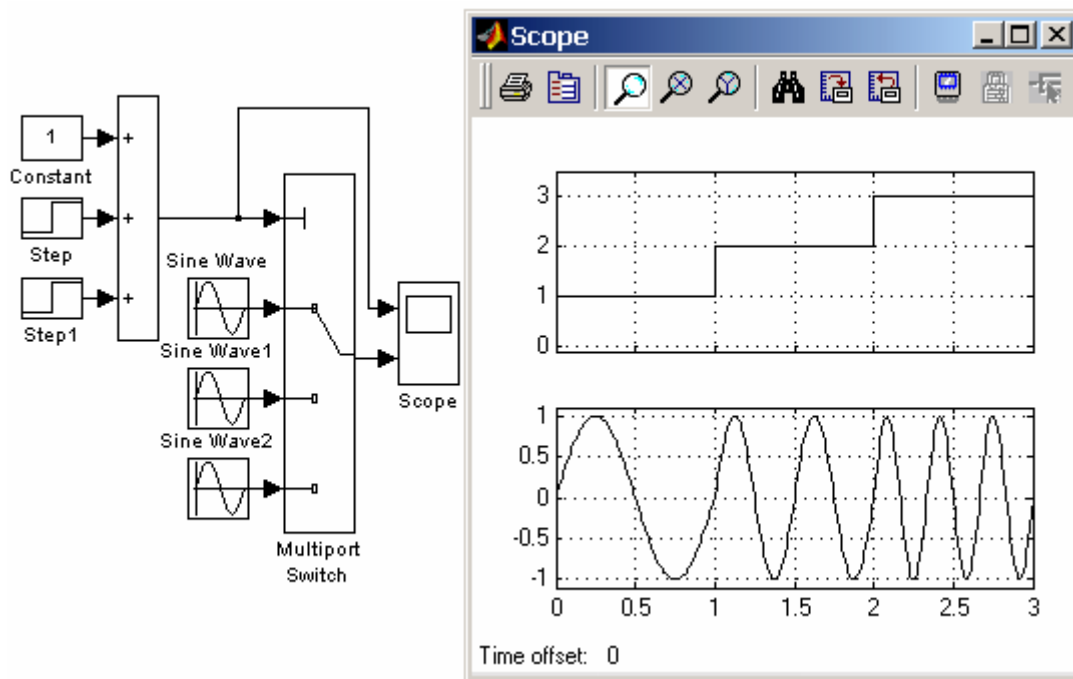


Рисунок 5.31 – Применение переключателя **Multiport Switch**

Количество входов блока **Multiport Switch** можно задать равным **1**. В этом случае на вход блока необходимо подать векторный сигнал, а сам блок будет пропускать на выход тот элемент вектора, номер которого совпадает с уровнем управляющего сигнала.

5.5.3. Блок ручного переключателя **Manual Switch**

Назначение:

Выполняет переключение входных сигналов по команде пользователя.

Параметры: Нет.

Командой на переключение является двойной щелчок левой клавишей мыши на изображении блока. При этом изображение блока изменяется, показывая, какой входной сигнал в данный момент проходит на выход блока. Переключение блока можно выполнять как до начала моделирования, так и в

процессе расчета.

На рисунке 5.32 показан пример использования блока **Manual Switch**.

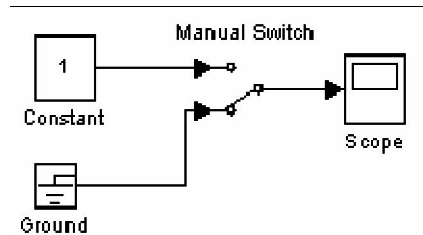


Рисунок 5.32 – Пример использования блока **Manual Switch**

5.6. Math – блоки математических операций

5.6.1. Блок вычисления модуля Abs

Назначение:

Выполняет вычисление абсолютного значения величины сигнала.

Параметры:

Saturate on integer overflow (флажок) – Подавлять переполнение целого. При установленном флажке ограничение сигналов целого типа выполняется корректно.

Пример использования блока **Abs**, вычисляющего модуль текущего значения синусоидального сигнала, показан на рисунке 5.33.

Блок **Abs** может использоваться также для вычисления модуля сигнала комплексного типа. На рисунке 5.34 показан пример вычисления модуля комплексного сигнала :

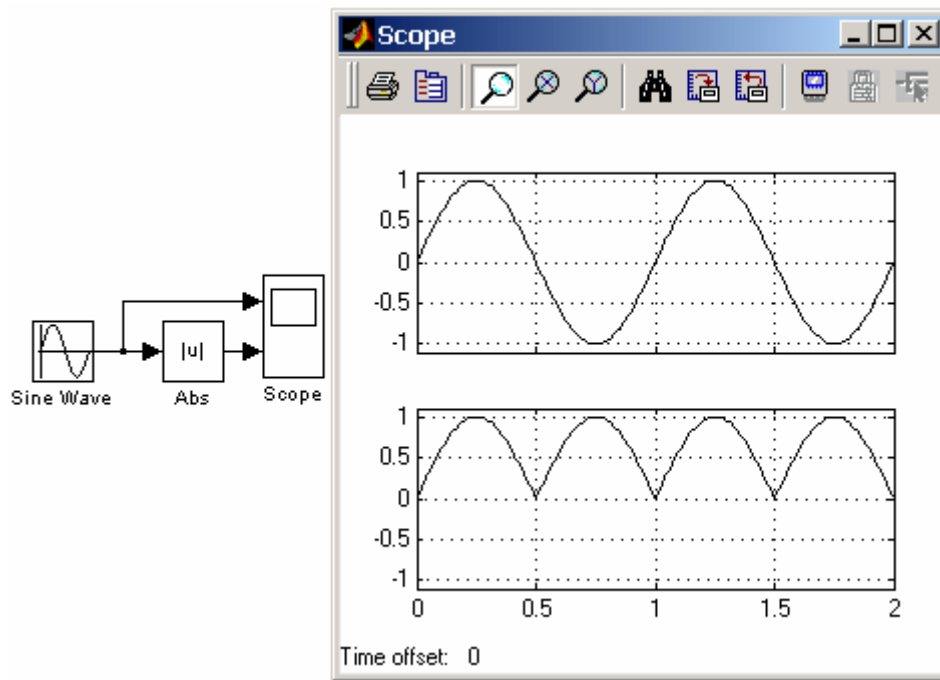


Рисунок 5.33 – Пример использования блока **Abs**

Модуль этого сигнала, как и следовало ожидать, равен **1** для любого момента времени.

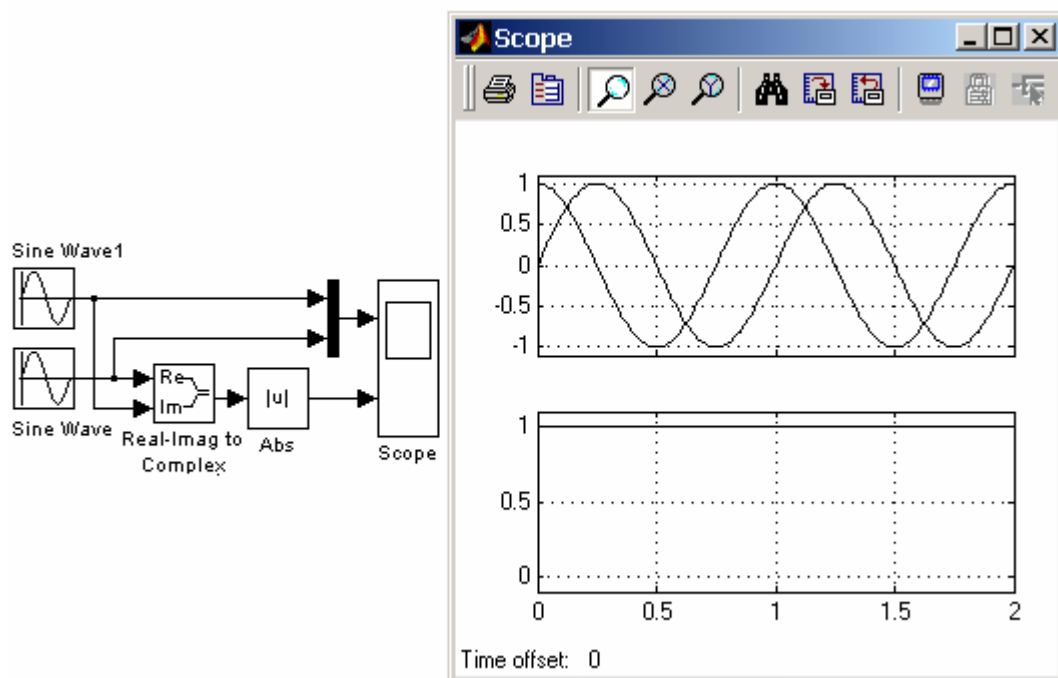


Рисунок 5.34 – Пример использования блока **Abs** для вычисления модуля комплексного сигнала

5.6.2. Блок вычисления суммы **Sum**

Назначение:

Выполняет вычисление суммы текущих значений сигналов.

Параметры:

1. **Icon shape** – Форма блока. Выбирается из списка.
 - **round** – окружность,
 - **rectangular** – прямоугольник.
2. **List of sign** – Список знаков. В списке можно использовать следующие знаки: + (плюс), - (минус) и | (разделитель знаков).
3. **Saturate on integer overflow** (флажок) – Подавлять переполнение целого. При установленном флажке ограничение сигналов целого типа выполняется корректно.

Количество входов и операция (сложение или вычитание) определяется списком знаков параметра **List of sign**, при этом метки входов обозначаются соответствующими знаками. В параметре **List of sign** можно также указать число входов блока. В этом случае все входы будут суммирующими.

Если количество входов блока превышает три, то удобнее использовать блок **Sum** прямоугольной формы.

Блок может использоваться для суммирования скалярных, векторных или матричных сигналов. Типы суммируемых сигналов должны совпадать. Нельзя, например, подать на один и тот же суммирующий блок сигналы целого и действительного типов.

Если количество входов блока больше, чем один, то блок выполняет поэлементные операции над векторными и матричными сигналами. При этом количество элементов в матрице или векторе должно быть одинаковым.

Если в качестве списка знаков указать цифру **1** (один вход), то блок можно использовать для определения суммы элементов вектора.

Примеры использования блока **Sum** показаны на рисунке 5.35.

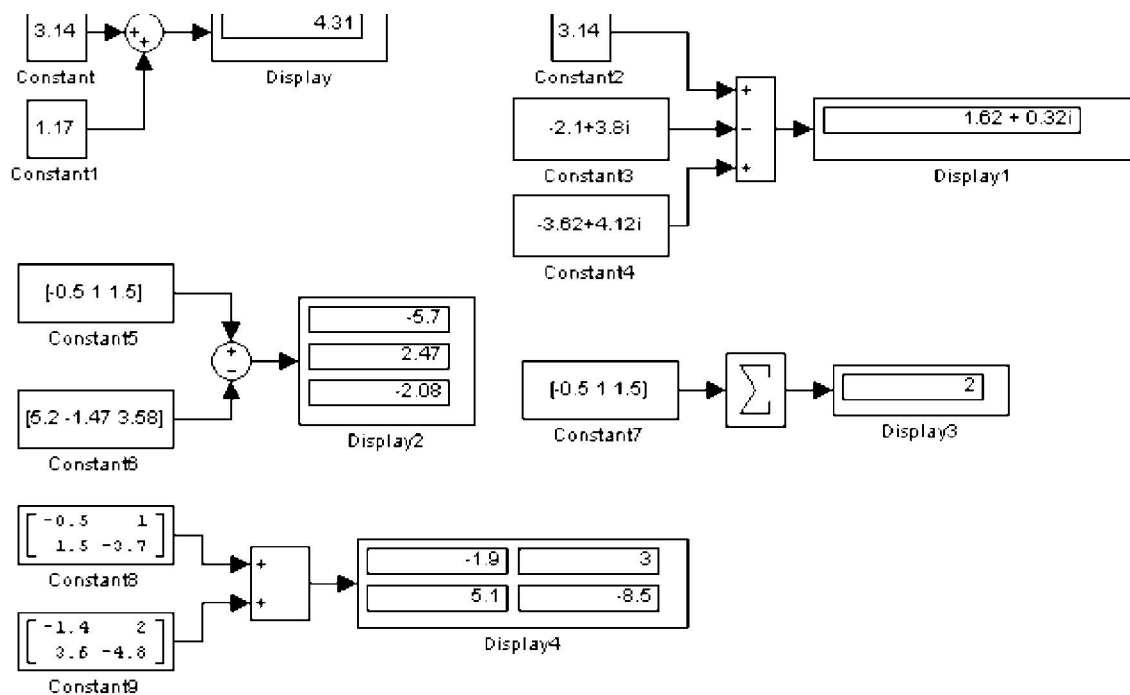


Рисунок 5.35 – Примеры использования блока Sum

5.6.3. Блок умножения Product

Назначение:

Выполняет вычисление произведения текущих значений сигналов.

Параметры:

1. **Number of inputs** – Количество входов. Может задаваться как число или как список знаков. В списке знаков можно использовать знаки «×» (умножить) и «/» (разделить).
2. **Multiplication** – Способ выполнения операции. Может принимать значения (из списка):
 - **Element-wise** – Поэлементный.
 - **Matrix** – Матричный.
3. **Saturate on integer overflow** (флажок) – Подавлять переполнение целого. При установленном флажке ограничение сигналов целого типа выполняется корректно. Если параметр **Number of inputs** задан списком, включающим кроме знаков умножения также знаки деления, то метки входов будут обозначены символами соответствующих операций.

Блок может использоваться для операций умножения или деления скалярных векторных или матричных сигналов. Типы входных сигналов блока должны совпадать. Если в качестве количества входов указать цифру **1** (один вход), то блок можно использовать для определения произведения элементов вектора.

Примеры использования блока **Product** при выполнении скалярных и поэлементных операций показаны на рисунке 5.36.

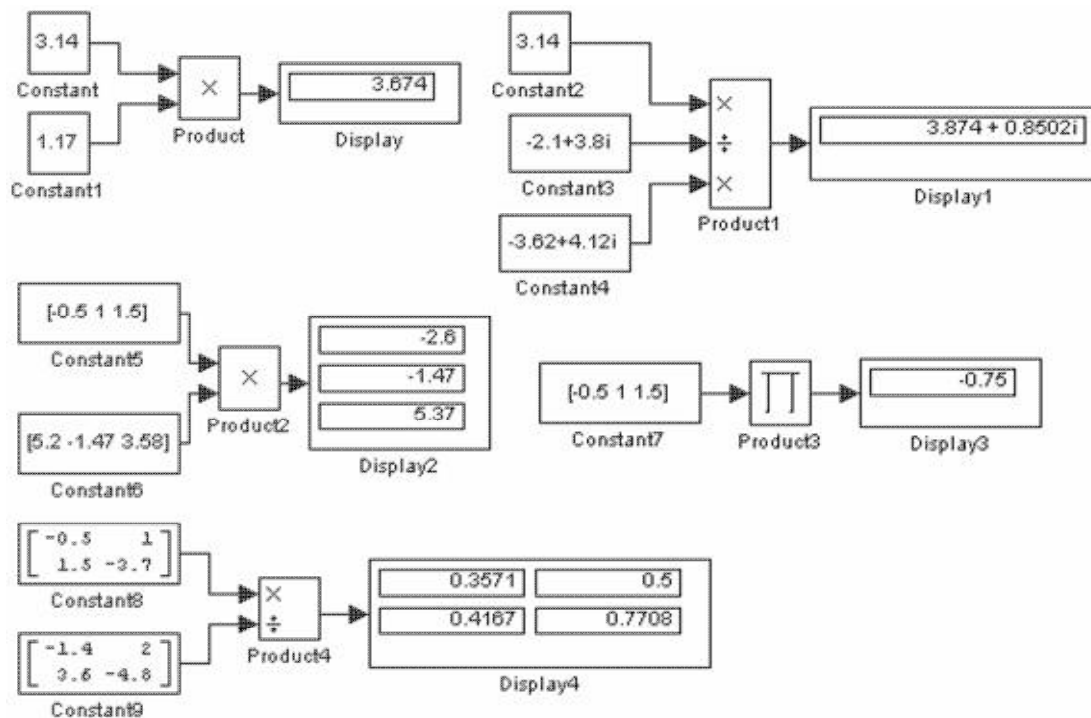


Рисунок 5.36 – Примеры использования блока **Product** при выполнении скалярных и поэлементных операций

При выполнении матричных операций необходимо соблюдать правила их выполнения. Например, при умножении двух матриц необходимо, чтобы количество строк первой матрицы равнялось количеству столбцов второй матрицы. Примеры использования блока **Product** при выполнении матричных операций показаны на рисунке 5.37. В примере показаны операции формирования обратной матрицы, деление матриц, а также умножение матриц.

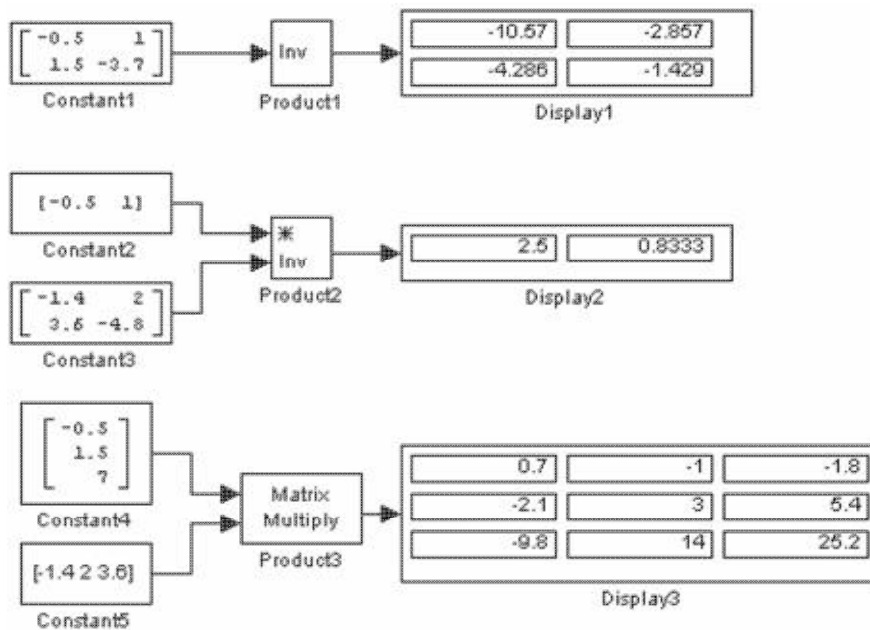


Рисунок 5.37 – Примеры использования блока **Product** при выполнении матричных операций

5.6.4. Блок определения минимального или максимального значения MinMax

Назначение:

Определяет максимальное или минимальное значение из всех сигналов, поступающих на его входы.

Параметры:

1. **Function** – Выходной параметр. Выбирается из списка:

- **min** – Минимальное значение.
- **max** – Максимальное значение.

2. **Number of input ports** – Количество входных портов.

Входные сигналы блока могут быть скалярными или векторными. Блок определяет максимальное или минимальное значение из всех скалярных сигналов, поступающих на его входы. Если входные сигналы являются векторными, то блок выполняет поэлементную операцию поиска минимального или максимального значения. В этом случае размерности векторов должны совпадать. Если количество входных портов блока задано равным **1**, то блок может использоваться для нахождения минимального или

максимального значения во входном векторе.

Примеры использования блока **MinMax** показаны на рисунке 5.38.

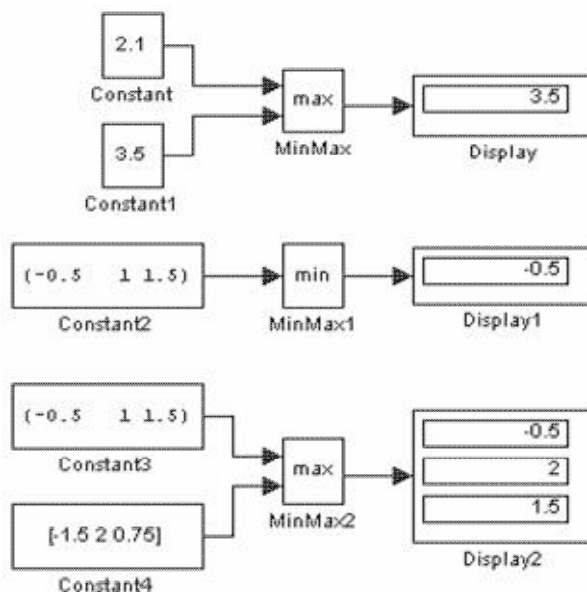


Рисунок 5.38 – Примеры использования блока **MinMax**

5.7. Signal&Systems - блоки преобразования сигналов и вспомогательные блоки

5.7.1. Мультиплексор (смеситель) Mux

Назначение: Объединяет входные сигналы в вектор.

Параметры:

1. **Number of Inputs** – Количество входов.
2. **Display option** – Способ отображения. Выбирается из списка:
 - **bar** – Вертикальный узкий прямоугольник черного цвета.
 - **signals** – Прямоугольник с белым фоном и отображением меток входных сигналов.
 - **none** – Прямоугольник с белым фоном без отображения меток входных сигналов.

Входные сигналы блока могут быть скалярными и (или) векторными.

Если среди входных сигналов есть векторы, то количество входов можно задавать как вектор с указанием числа элементов каждого вектора.

Например, выражение $[2 \ 3 \ 1]$ определяет три входных сигнала, первый сигнал – вектор из двух элементов, второй сигнал – вектор из трехэлементов, и последний сигнал – скаляр. В том случае, если размерность входного вектора не совпадает с указанной в параметре **Number of Inputs**, после начала расчета **Simulink** выдаст сообщение об ошибке. Размерность входного вектора можно задавать как **-1** (минус один). В этом случае размерность входного вектора может быть любой.

Параметр **Number of Inputs** можно задавать также в виде списка меток сигналов, например: **Vector1, Vector2, Scalar**. В этом случае метки сигналов будут отображаться рядом с соответствующими соединительными линиями.

Сигналы, подаваемые на входы блока должны быть одного типа (действительного или комплексного).

Примеры использования блока **Mux** показаны на рисунке 5.39.

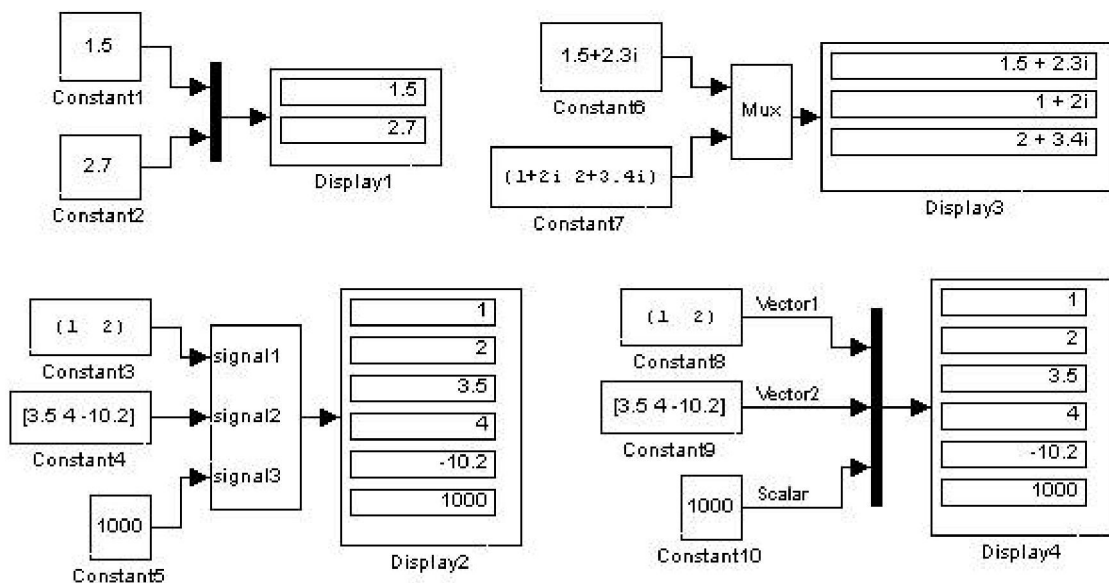


Рисунок 5.39 – Примеры использования блока **Mux**

5.7.2. Демультимплексор (разделитель) **Demux**

Назначение: Разделяет входной векторный сигнал на отдельные составляющие.

Параметры:

1. **Number of Outputs** – Количество выходов.
2. **Bus Selection Mode** (флажок) – Режим разделения векторных сигналов.

Входным сигналом в обычном режиме является вектор, сформированный любым способом. Выходными сигналами являются скаляры или векторы, количество которых и размерность определяется параметром **Number of Outputs** и размерностью входного вектора.

Если количество выходов **P** (значение параметра **Number of Outputs**) равно размерности входного сигнала **N**, то блок выполняет разделение входного вектора на отдельные элементы.

Если количество выходов **P** меньше, чем размерность входного сигнала **N**, то размерность первых **P-1** выходных сигналов равна отношению N/P , округленному до ближайшего большего числа, а размерность последнего выходного сигнала равна разности между размерностью входного сигнала и суммой размерностей первых **P-1** выходов. Например, если размерность входного сигнала равна восьми, а количество выходов равно трем, то первые два выходных вектора будут иметь размерность $\text{ceil}(8/3) = 3$, а последний выходной вектор будет иметь размерность $8 - (3+3) = 2$.

Параметр **Number of Outputs** может быть задан также с помощью вектора, определяющего размерность каждого выходного сигнала. Например, выражение **[2 3 1]** определяет три выходных сигнала, первый сигнал – вектор из двух элементов, второй сигнал – вектор из трех элементов, и последний сигнал – скаляр. Размерность можно также задавать как **-1** (минус один). В этом случае размерность соответствующего выходного сигнала определяется как разность между размерностью входного вектора и суммой размерностей заданных выходных сигналов. Например, если размерность входного вектора равна шести, а параметр **Number of Outputs** задан выражением **[1 -1 3]**, то второй выходной сигнал будет иметь размерность $6 - (3+1) = 2$.

Примеры использования блока **Demux** показаны на рисунке 5.40.

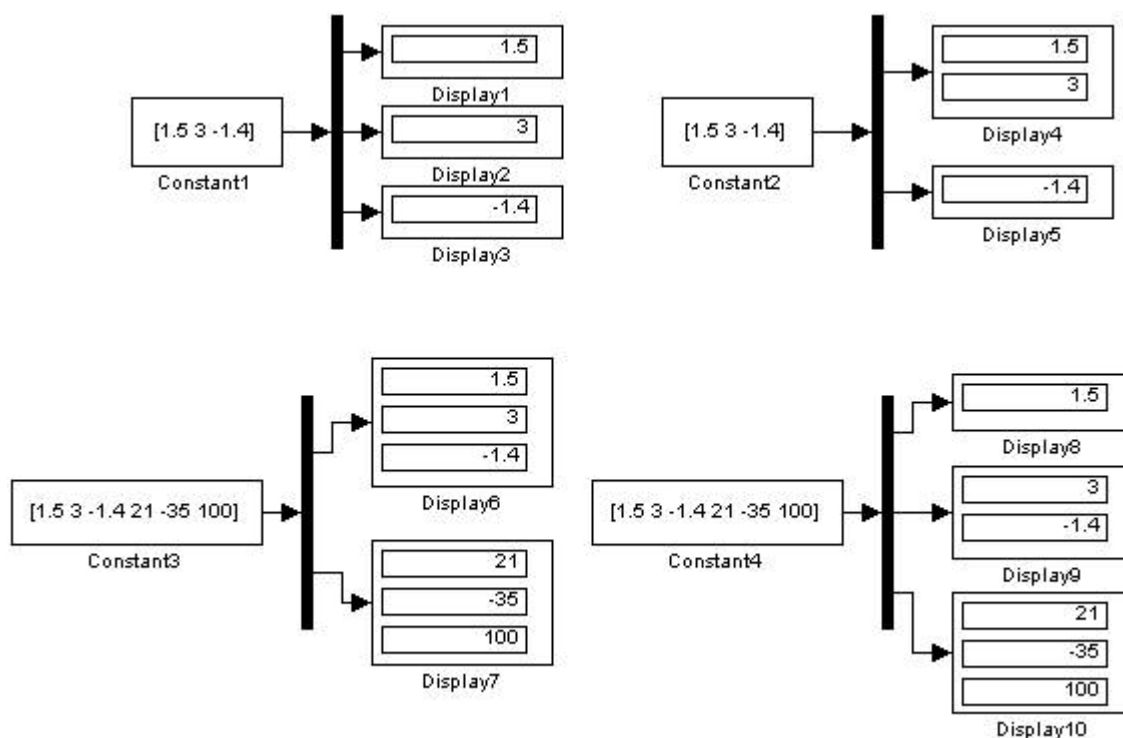


Рисунок 5.40 – Примеры использования блока **Demux**

В режиме **Bus Selection Mode** блок **Demux** работает не с отдельными элементами векторов, а с векторными сигналами в целом. Входной сигнал в этом режиме должен быть сформирован блоком **Mux** или другим блоком **Demux**. Параметр **Number of Outputs** в этом случае задается в виде скаляра, определяющего количество выходных сигналов, либо в виде вектора, каждый элемент которого определяет количество векторных сигналов в данном выходном сигнале. Например, при входном сигнале, состоящем из трех векторов параметр **Number of Outputs**, заданный вектором **[2 1]**, определит два выходных сигнала, первый из которых будет содержать два векторных сигнала, а второй – один.

Примеры использования блока **Demux** в режиме **Bus Selection Mode** показаны на рисунке 5.41.

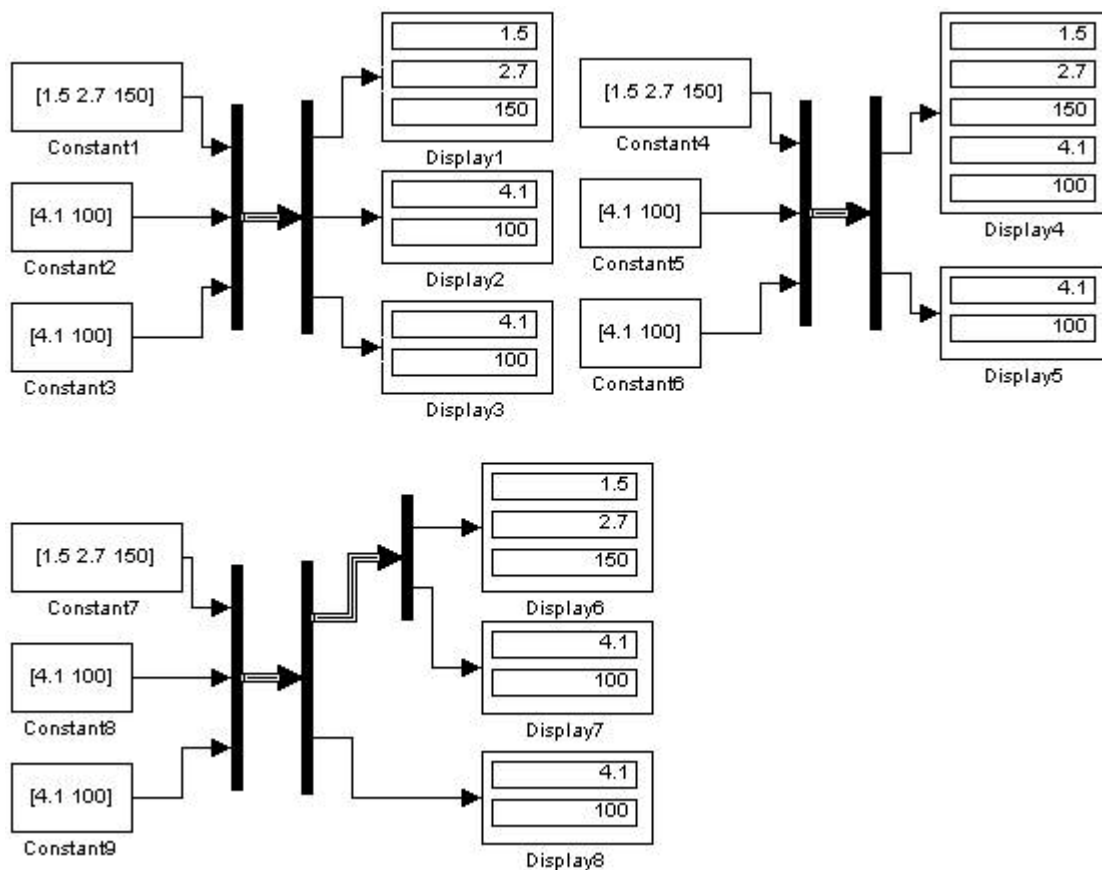


Рисунок 5.41 – Примеры использования блока **Demux** в режиме **Bus Selection Mode**

5.8. Subsystem – подсистемы.

Подсистема – это фрагмент **Simulink**-модели, оформленный в виде отдельного блока. Использование подсистем при составлении модели имеет следующие положительные стороны:

1. Уменьшает количество одновременно отображаемых блоков на экране, что облегчает восприятие модели (в идеале модель полностью должна отображаться на экране монитора).
2. Позволяет создавать и отлаживать фрагменты модели по отдельности, что повышает технологичность создания модели.
3. Позволяет создавать собственные библиотеки.
4. Дает возможность синхронизации параллельно работающих подсистем.
5. Позволяет включать в модель собственные справочные средства.
6. Дает возможность связывать подсистему с каким-либо **m**-файлом, обеспечивая запуск этого файла при открытии подсистемы (нестандартное

открытие подсистемы).

Использование подсистем и механизма их блоков позволяет создавать блоки, не уступающие стандартным по своему оформлению (собственное окно параметров блока, пиктограмма, справка и т.п.).

Количество подсистем в модели не ограничено, кроме того, подсистемы могут включать в себя другие подсистемы. Уровень вложенности подсистем друг в друга также не ограничен.

Связь подсистемы с моделью (или подсистемой верхнего уровня иерархии) выполняется с помощью входных (блок **Inport** библиотеки **Sources**) и выходных (блок **Outport** библиотеки **Sinks**) портов. Добавление в подсистему входного или выходного порта приводит к появлению на изображении подсистемы метки порта, с помощью которой внешние сигналы передаются внутрь подсистемы или выводятся в основную модель. Переименование блоков **Inport** или **Outport** позволяет изменить отображаемые на пиктограмме подсистемы со стандартных (**In** и **Out**) на те, которые нужны пользователю.

Подсистемы могут быть виртуальными (**Subsystem**) и монолитными (**Atomic Subsystem**). Отличие этих видов подсистем заключается в порядке выполнения блоков во время расчета. Если подсистема является виртуальной, то **Simulink** игнорирует наличие границ отделяющих такую подсистему от модели при определении порядка расчета блоков. Иными словами, в виртуальной системе сначала могут быть рассчитаны выходные сигналы нескольких блоков, затем выполнен расчет блоков в основной модели, а затем вновь выполнен расчет блоков входящих в подсистему. Монолитная подсистема считается единым (неделимым) блоком и **Simulink** выполняет расчет всех блоков в такой подсистеме, не переключаясь на расчеты других блоков в основной модели. Изображение монолитной подсистемы имеет более толстую рамку по сравнению с виртуальной подсистемой. Подсистемы могут быть также управляемыми или

неуправляемыми. Управляемые подсистемы всегда являются монолитными. Управляемые подсистемы имеют дополнительные (управляющие) входы, на которые поступают сигналы, активизирующие данную подсистему. Управляющие входы расположены сверху или снизу подсистемы. Когда управляемая подсистема активизирована – она выполняет вычисления. В том случае если управляемая подсистема пассивна, она не выполняет вычисления, а значения сигналов на ее выходах определяются настройками выходных портов.

Для создания в модели подсистемы можно воспользоваться двумя способами:

1. Скопировать нужную подсистему из библиотеки **Subsystem** в модель.
2. Выделить с помощью мыши нужный фрагмент модели и выполнить команду **Create Subsystem** из меню **Edit** окна модели. Выделенный фрагмент будет помещен в подсистему, а входы и выходы подсистемы будут снабжены соответствующими портами. Данный способ позволяет создать виртуальную неуправляемую подсистему. В дальнейшем, если это необходимо, можно сделать подсистему монолитной, изменив ее параметры, или управляемой, добавив управляющий элемент из нужной подсистемы находящейся в библиотеке. Отменить группировку блоков в подсистему можно командой **Undo**.

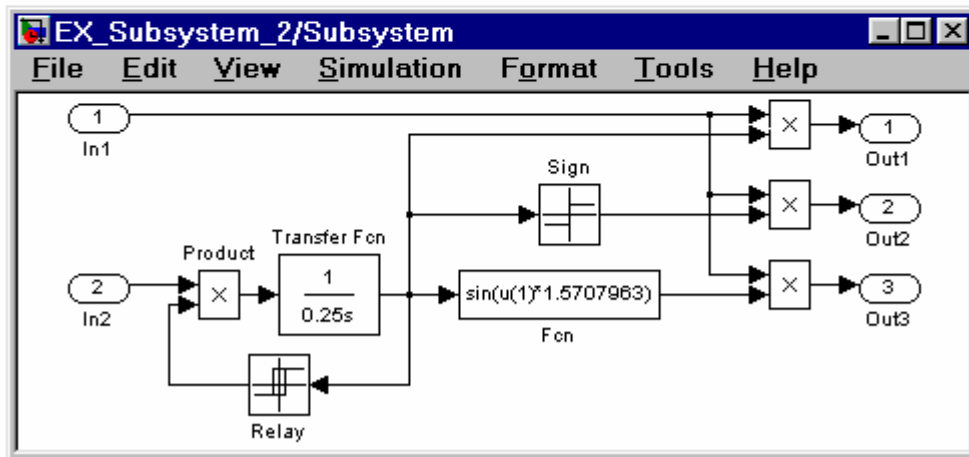
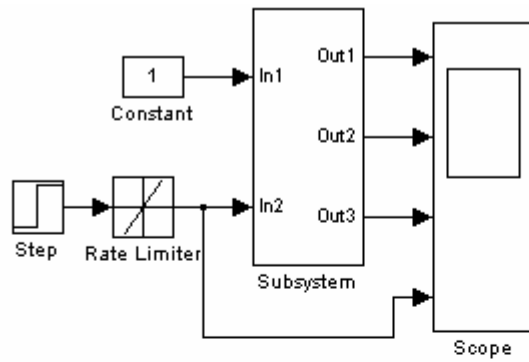


Рисунок 5.42 – Модель, использующая подсистему

Лабораторная работа №1.

МАТЕМАТИЧЕСКИЕ МОДЕЛИ ЛИНЕЙНЫХ СИСТЕМ АВТОМАТИЧЕСКОГО УПРАВЛЕНИЯ

Цель работы: Ознакомиться со средой имитационного моделирования Simulink. Освоить основные приемы моделирования систем автоматического управления.

Краткие теоретические сведения

Поведение динамической линейной системы автоматического управления может быть описано скалярным дифференциальным уравнением n -го порядка

$$y^{(n)} + a_{n-1}y^{(n-1)} + \dots + a_1y^{(1)} + a_0y = b_m u^{(m)} + b_{m-1}u^{(m-1)} + \dots + b_1u^{(1)} + b_0u, \quad (1.1)$$

где y – выходная переменная; u – входной сигнал; m – порядок производной входного сигнала; a_i и b_j – постоянные коэффициенты.

При условии, что $m < n$, уравнение (1.1) можно записать в виде системы уравнений первого порядка

$$\begin{cases} x_1^{(1)} = \alpha_{11} \cdot x_1 + \alpha_{12} \cdot x_2 + \dots + \alpha_{1n} \cdot x_n + \beta_1 \cdot u, \\ x_2^{(1)} = \alpha_{211} \cdot x_1 + \alpha_{22} \cdot x_2 + \dots + \alpha_{2n} \cdot x_n + \beta_2 \cdot u, \\ \dots \\ x_n^{(1)} = \alpha_{n1} \cdot x_1 + \alpha_{n2} \cdot x_2 + \dots + \alpha_{nn} \cdot x_n + \beta_n \cdot u, \\ y = C_1 \cdot x_1 + C_2 \cdot x_2 + \dots + C_n \cdot x_n, \end{cases} \quad (1.2)$$

где x_i – координаты вектора состояния, α_{ii} и β_i – постоянные коэффициенты.

Система уравнений (1.2) может быть представлена в компактной векторно-матричной форме

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned} \quad (1.3)$$

где A – $n \times n$ – мерная матрица постоянных коэффициентов системы; B – $n \times 1$ – мерная матрица постоянных коэффициентов входа; C – $1 \times n$ – мерная матрица постоянных коэффициентов выхода; X – n -мерный вектор состояния.

Порядок выполнения работы

1. Ознакомиться с пакетом прикладных программ MATLAB-Simulink.
2. В соответствии с вариантом задания (см. табл. 6.1) построить схему моделирования линейной системы автоматического управления, используя уравнение (1.1.-1.3).
3. Осуществить моделирование системы при двух видах входных воздействий: $u=1(t)$ и $u=\sin(t)$. Начальные условия нулевые. На монитор выводить графики сигналов $y(t)$ и $u(t)$. Продолжительности интервалов наблюдения выбрать самостоятельно.
4. Осуществить моделирование свободного движения системы с нулевыми и ненулевыми начальными условиями (см. табл.6.2). Снять выходные характеристики $y(t)$ системы автоматического управления. Получить фазовый портрет.

Таблица 1.1 – Варианты параметров моделей

Вариант	1	2	3	4	5	6	7	8	9	10	11	12
порядок модели n	3	3	3	3	3	3	2	2	2	2	2	2
a_0	9	5	5	8	7	15	7	2	1	25	30	0,12
a_1	6	4	4	6	5	5	3	0,5	0,5	1	0,8	1
a_2	3	3	2	2	10	-	-	-	-	-	-	-
b_0	12	2,5	7,5	12	10	15	10	4	2	25	30	0,1
b_1	2	2	0	1	3	0,5	6	2	2	2	3	2
b_2	0,1	3	5	10	1,5	1	0	0	0	0	0	0

Таблица 6.2 – Варианты начальных условий моделей

Вариант	1	2	3	4	5	6	7	8	9	10	11	12
Порядок модели n	3	3	3	3	3	3	2	2	2	2	2	2
$y(0)$	1	1	1	1	1	1	1	1	1	1	1	1
$y^{(1)}(0)$	0,5	-0,2	-0,4	0,1	-0,5	0,5	0,4	1	-0,5	0	0,5	0
$y^{(2)}(0)$	0	0,1	0,2	-0,1	0	0,1	-	-	-	-	-	-

Вопросы для самоконтроля

1. Составьте схему моделирования уравнения $y^{(1)} + 3y = 2u^{(1)} + 5u$.
2. Назовите виды математических моделей?
3. Почему для моделирования динамических систем используются блоки интегрирования?
4. Поясните принцип составления модели вход – выход.

Лабораторная работа № 2.

ТИПОВЫЕ ДИНАМИЧЕСКИЕ ЗВЕНЬЯ СИСТЕМ АВТОМАТИЧЕСКОГО УПРАВЛЕНИЯ

Цель работы: Исследование переходных характеристик и динамических свойств типовых звеньев систем автоматического управления.

Краткие теоретические сведения

Типовыми динамическими звеньями называются простейшие составные части систем автоматического управления, поведение которых описывается обыкновенными дифференциальными уравнениями не выше 2-ого порядка:

$$a_2 y^{(2)} + a_1 y^{(1)} + a_0 y = b_1 u^{(1)} + b_0 u, \quad (2.1)$$

где y и u – соответственно выходная переменная и управляющее воздействие звена; a_i и b_i – постоянные коэффициенты.

С использованием оператора дифференцирования $p = d / dt$ уравнение (2.1) имеет вид:

$$a_2 p^2 y + a_1 p y + a_0 y = b_1 p u + b_0 u, \quad (2.2)$$

Определяем передаточную функцию $W(p)$ звена, учитывая при этом, что начальные условия для уравнения (2.2) нулевые

$$W(p) = \frac{y}{u} = \frac{b_1 p + b_0}{a_2 p^2 + a_1 p + a_0} \quad (2.3)$$

Динамические свойства звеньев определяются по их реакции на типовое входное воздействие. Наиболее простым типовым воздействием является единичная ступенчатая функция $1(t)$, удовлетворяющая условиям

$$1(t) = \begin{cases} 0, & \text{если } t < 0 \\ 1, & \text{если } t \geq 0 \end{cases} \quad (2.4)$$

Одной из реакций звена является переходная функция $h(t)$ – изменение выходной переменной во времени при подаче на вход звена единичной ступенчатой функции $1(t)$. Переходная функция характеризует переход звена (системы) от одного равновесного состояния или установившегося режима к другому.

По графику $h(t)$ можно определить математическую модель исследуемого динамического звена и его параметры.

Интегрирующее звено

Описывается уравнениями:

$$y^{(1)} = k u \quad \text{или} \quad y = \frac{k}{p} u, \quad (2.5)$$

где k – постоянный коэффициент.

Переходная функция звена

$$h(t) = k t \cdot 1(t) \quad (2.6)$$

Интегрирующее звено с замедлением

Описывается уравнениями

$$Ty^{(2)} + y^{(1)} = ku \text{ или } y = \frac{k}{p(Tp + 1)}u, \quad (2.7)$$

где T – постоянная времени.

Переходная функция звена

$$h(t) = \kappa [t - T(1 - e^{-\frac{t}{T}})] \cdot 1(t) \quad (2.8)$$

Графики переходных функций интегрирующих звеньев показаны на рисунке 7.1.

Изодромное звено

Описывается уравнениями

$$y^{(1)} = k(Tu^{(1)} + u) \text{ или } y = \frac{k(Tp + 1)}{p}u, \quad (2.9)$$

его переходная функция

$$h(t) = \kappa(t + T) \cdot 1(t).$$

Реальное дифференцирующее звено

Описывается уравнениями

$$Ty^{(1)} + y = ku^{(1)} \text{ или } y = \frac{kp}{Tp + 1}u, \quad (2.10)$$

его переходная функция

$$h(t) = \frac{\kappa}{T} e^{-\frac{t}{T}} \cdot 1(t). \quad (2.11)$$

Графики переходных функций изодромного и реального дифференцирующего звеньев изображены на рисунке 2.2.

Апериодическое звено первого порядка

Описывается уравнениями

$$Ty^{(1)} + y = ku \text{ или } y = \frac{k}{Tp + 1}u, \quad (2.12)$$

его переходная функция

$$h(t) = k(1 - e^{-\frac{t}{T}}) \cdot 1(t) \quad (2.13)$$

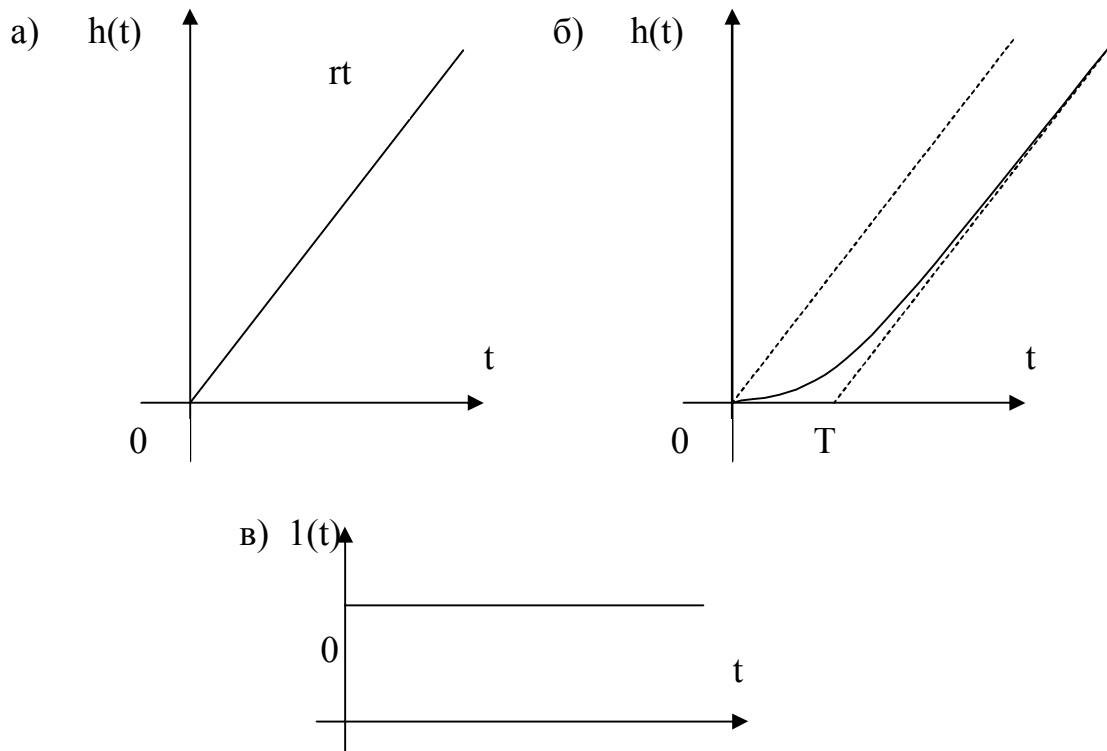


Рисунок 7.1 – Графики переходных функций интегрирующего звена (а) и интегрирующего звена с замедлением (б), единичная ступенчатая функция $I(t)$ - (в)

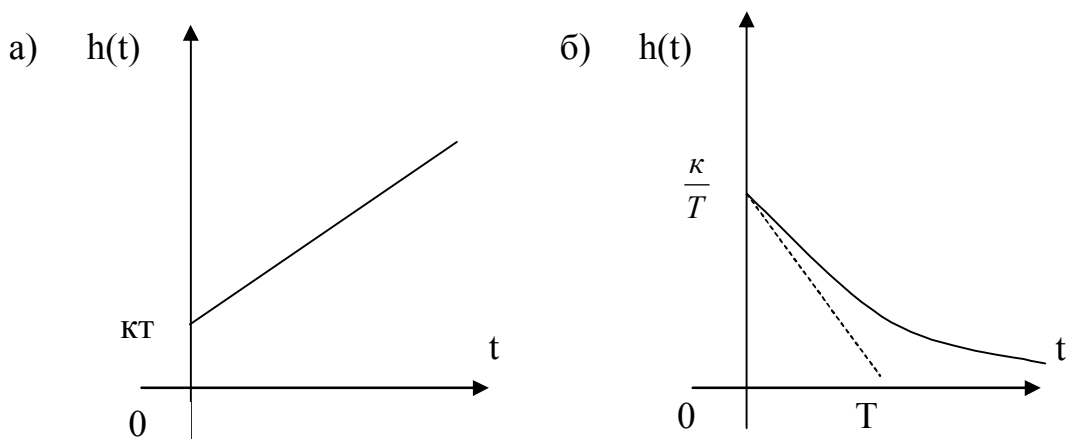


Рисунок 7.2 – Графики переходных функций издромного (а) и реального дифференцирующего (б) звеньев

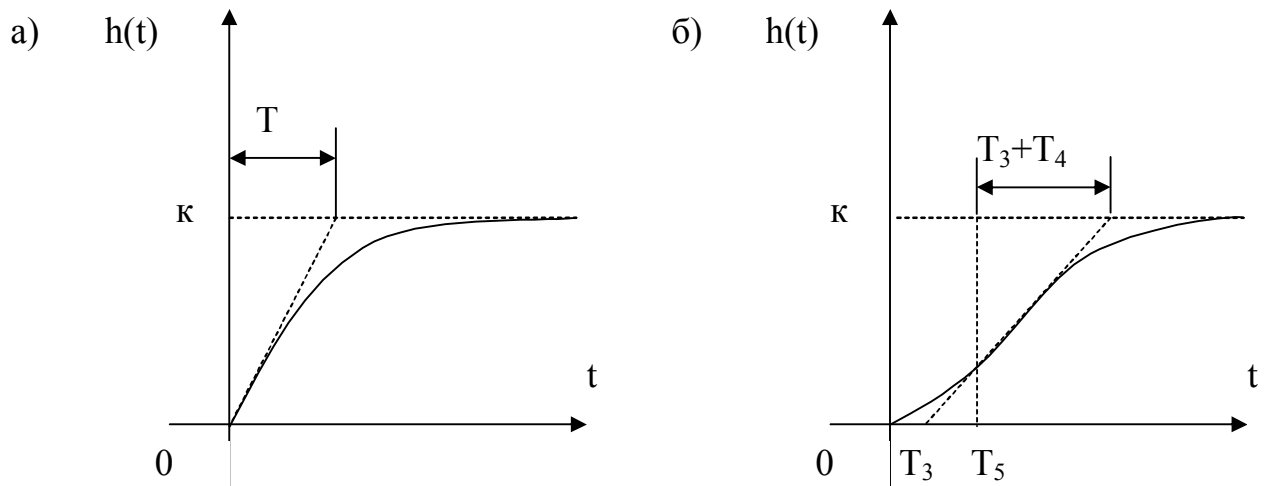


Рисунок 7.3 – Графики переходных функций аperiodических звеньев первого порядка (а) и второго порядка (б)

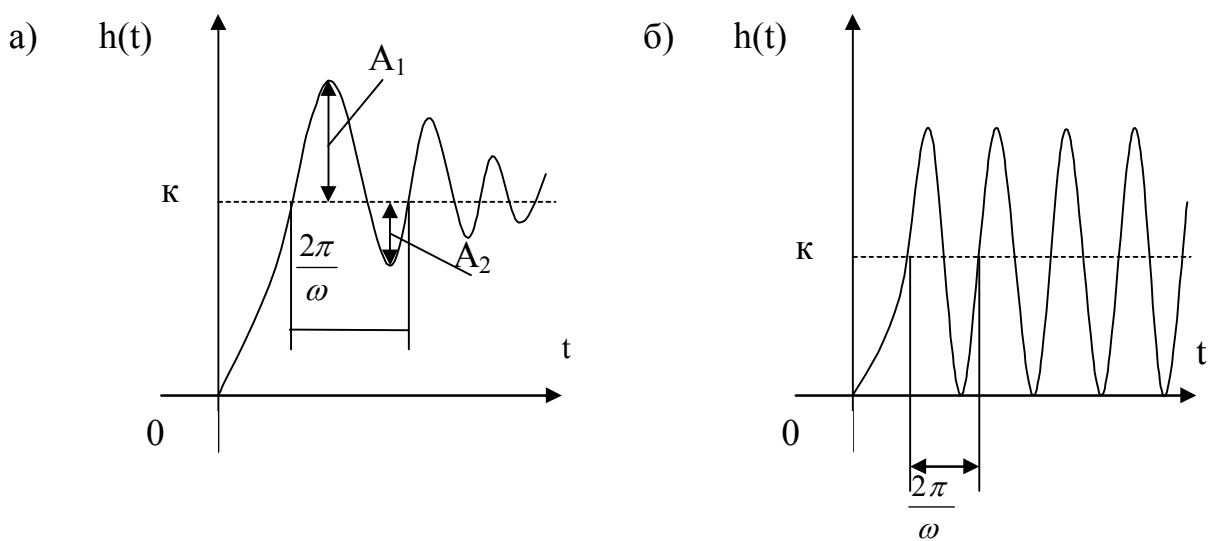


Рисунок 7.4 – Графики переходных функций колебательного (а) и консервативного (б) звеньев

Аperiodическое звено 2-ого порядка

Описывается уравнениями

$$T_2^2 y^{(2)} + T_1 y^{(1)} + y = ku \quad \text{или} \quad y = \frac{k}{T_2^2 p^2 + T_1 p + 1} u, \quad (2.14)$$

где T_1, T_2 – постоянные времени ($T_1 > 2T_2$). При этом корни характеристического уравнения $T_2^2 p^2 + T_1 p + 1$ являются вещественными и

отрицательными. Знаменатель передаточной функции аperiodического звена 2-ого порядка может быть разложен на множители

$$y = \frac{\kappa}{(T_3 p + 1)(T_4 p + 1)} u, \quad (2.15)$$

где $T_3 = \frac{T_1}{2} + \sqrt{\frac{T_1^2}{4} - T_2^2}$; $T_4 = \frac{T_1}{2} - \sqrt{\frac{T_1^2}{4} - T_2^2}$.

В связи с этим, аperiodическое звено второго порядка эквивалентно двум аperiodическим звеньям первого порядка, соединенным последовательно между собой и имеющим коэффициент усиления κ и постоянные времени T_3 и T_4 .

Переходная функция аperiodического звена второго порядка имеет вид

$$h(t) = \kappa \left(1 - \frac{T_3}{T_3 - T_4} e^{-\frac{t}{T_3}} + \frac{T_4}{T_3 - T_4} e^{-\frac{t}{T_4}} \right) \cdot 1(t) \quad (2.16)$$

Графики переходных функций аperiodических звеньев показаны на рисунке. 7.3, а.

$$T_5 = \frac{T_3 T_4}{T_3 + T_4} \ln \left(\frac{T_3}{T_4} \right)$$

Колебательное звено

Описывается дифференциальным уравнением, что и аperiodическое звено второго порядка. Однако корни характеристического уравнения являются комплексными. Уравнение и передаточная функция колебательного звена представляются в виде

$$T^2 y^{(2)} + 2\xi T y^{(1)} + y = \kappa u, \quad (2.17)$$

$$W(p) = \frac{\kappa}{T^2 p^2 + 2\xi T p + 1} = \frac{\kappa \omega_0^2}{p^2 + 2\xi \omega_0 p + \omega_0^2}, \quad (2.18)$$

где $\omega_0 = 1/T$ – частота свободных колебаний при отсутствии затухания; ζ – коэффициент затухания ($0 < \zeta < 1$).

Переходная функция колебательного звена:

$$h(t) = \kappa \left[1 - e^{-\gamma t} \left(\cos \omega t + \frac{\gamma}{\omega} \sin \omega t \right) \right] \cdot 1(t), \quad (2.19)$$

где $\gamma = \frac{\omega}{\pi} \ln \frac{A_1}{A_2}$, $\gamma = \xi \omega_0$, $\omega = \omega_0 \sqrt{1 - \xi^2}$.

Параметры выражения (2.19) можно легко определить по графику переходной функции (см. рисунок 7.4 а).

Консервативное звено

Может быть получено из колебательного звена, если $\zeta=0$. В этом случае корни характеристического уравнения $T^2 p^2 + 1 = 0$ будут чисто мнимые.

Передаточная функция консервативного звена имеет вид:

$$W(p) = \frac{k}{T^2 p^2 + 1}, \quad (2.20)$$

а его переходная функция

$$h(t) = k(1 - \cos \omega_0 t) \cdot 1(t) \quad (2.21)$$

Графики переходных функций колебательного и консервативного звеньев показаны на рисунке 7.4.

Порядок выполнения работы

1. Построить схемы моделирования динамических звеньев:

- интегрирующего;
- интегрирующего с запаздыванием;
- издромного;
- реального дифференцирующего
- апериодического первого порядка;
- апериодического второго порядка;
- колебательного;
- консервативного.

Параметры звеньев установить в соответствии с вариантом задания (см. табл. 7.1.)

2. Осуществить моделирование и снять переходные характеристики типовых динамических звеньев.

3. Для колебательного звена определить значение коэффициента затухания ζ , при котором время переходного процесса будет минимальным.

4. Сделать сравнительный анализ результатов моделирования.

Таблица 7.1 – Параметры динамических звеньев

Варианты	1	2	3	4	5	6	7	8	9	10	11	12
k	5	1	10	4	1,5	3	10	14	5	4	1	2
T	2	0,1	5	1	0,2	3	5	5	1	0,4	0,2	0,2
T_1	3	0,2	4	2	0,4	6	3	6	2	0,2	0,3	0,25
T_2	1,3	0,8	1	0,8	0,45	1	1,1	1,41	1	0,3	0,35	0,3
ζ	0,2	0,3	0,4	0,5	0,25	0,2	0,25	0,3	0,4	0,6	0,5	0,55

Вопросы для самоконтроля

1. Назовите типовое динамическое звено, у которого корни знаменателя передаточной функции чисто мнимые, а числитель передаточной функции вещественная постоянная величина?

2. Какому динамическому звену соответствует переходная функция $h(t) = (1 - 2e^{-0.5t} + e^{-t}) \cdot 1(t)$.

Определить параметры этого звена.

3. Динамическое звено описывается дифференциальным уравнением $4y^{(2)} + ay^{(1)} + y = 3u$.

Найти значение параметра a , при котором звено будет колебательным.

4. Нарисовать электрическую схему дифференцирующего звена.

5. Определить переходную функцию $h(t)$ динамического звена, заданного уравнением: $y^{(1)} + 2y = 1.5u$.

Лабораторная работа № 3.

АНАЛИЗ ТОЧНОСТИ СИСТЕМ АВТОМАТИЧЕСКОГО УПРАВЛЕНИЯ

Цель работы: Исследование точности систем автоматического регулирования в различных типовых режимах.

Краткие теоретические сведения

Точность работы любой системы автоматического управления наиболее полно характеризуется мгновенным значением *ошибки рассогласования* $\varepsilon(t)$, равной разности между заданной $g(t)$ и действительной $y(t)$ значениями регулируемой переменной в соответствии с уравнением

$$\varepsilon(t) = g(t) - y(t). \quad (3.1)$$

При этом значение $\varepsilon(t)$ оценивается при типовых входных воздействиях: постоянном, линейно или квадратично нарастающим.

Для характеристики точностных свойств систем управления используется понятие установившейся ошибки слежения. Установившаяся ошибка $\varepsilon_y(t)$, представляет собой функцию времени, удовлетворяющую условию

$$\lim_{t \rightarrow \infty} [\varepsilon(t) - \varepsilon_y(t)] = 0 \quad (3.2)$$

для любых начальных условий и заданного воздействия, т.е. она характеризует ошибку слежения, установившуюся после завершения переходного процесса. Предельное значение установившейся ошибки определяется выражением:

$$\varepsilon = \lim_{t \rightarrow \infty} \varepsilon(t) \quad (3.3)$$

Величина предельного значения установившейся ошибки при типовом воздействии (табл. 3.1) наиболее просто может быть рассчитана, если использовать передаточную функцию замкнутой системы по ошибке рассогласования:

$$\Phi_{\varepsilon}(p) = \frac{E(p)}{G(p)} = \frac{1}{1+W(p)}, \quad (3.4)$$

где $E(p)$ и $G(p)$ – соответственно изображения величины рассогласования и задающего воздействия; $W(p)$ – передаточная функция разомкнутой системы, включающая в себя передаточные функции объекта регулирования $W_0(p)$ и регулятора $R(p)$ (рисунок 8.1).

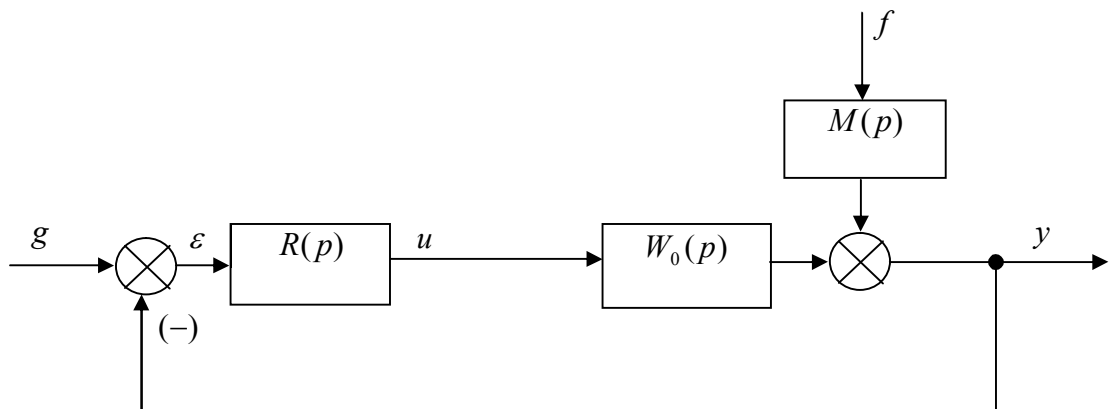


Рисунок. 8.1 – Система автоматического управления:

u – управление; f – возмущающее воздействие; $M(p)$ – передаточная функция для введения в систему возмущения

Значение установившейся ошибки определяется согласно теореме о конечном значении:

$$\varepsilon_y = \lim_{p \rightarrow 0} p \cdot \Phi_{\varepsilon}(p) \cdot G(p) \quad (3.5)$$

Разложим $\Phi_{\varepsilon}(p)$ в ряд Тейлора в окрестности точки $p=0$

$$\Phi_{\varepsilon}(p) = C_0 + C_1 p + \frac{C_2}{2!} p^2 + \frac{C_3}{3!} p^3 + \dots, \quad (3.6)$$

где $c_i = \left[\frac{d^i}{dp^i} \phi_\varepsilon \right]_{p=0}, i = 0, 1, 2, \dots$

Подставляя (3.6) в (3.5) и переходя во временную область, получаем выражение установившейся ошибки при произвольном входном воздействии

$$\varepsilon_y(t) = c_0 g(t) + c_1 \frac{d}{dt} g(t) + \frac{c_2}{2!} \frac{d^2}{dt^2} g(t) + \frac{c_3}{3!} \frac{d^3}{dt^3} g(t) + \dots \quad (3.7)$$

Здесь постоянные c_i носят название *коэффициентов ошибок*. Если $g(t)$ изменяется *достаточно медленно*, то для приближенной оценки $\varepsilon_y(t)$ можно использовать конечное число членов ряда (3.7).

Точность работы системы связана с *порядком астатизма*. Система называется *статической*, если она имеет нулевой порядок астатизма, т.е. в выражении (3.7) $c_0 \neq 0$. В общем случае, если система имеет k -й порядок астатизма, то $c_i = 0$ для всех $0 \leq i < k$ и $c_k \neq 0$.

Таблица 8.1 – Типовые задающие воздействия

Изображение типового задающего воздействия	Постоянное $g(t) = A$	Линейно нарастающее $g(t) = V \cdot t$	Квадратично нарастающее $g(t) = \frac{at^2}{2}$
$G(p)$	$\frac{A}{P}$	$\frac{V}{P^2}$	$\frac{a}{p^3}$

Порядок астатизма системы управления устанавливается на основе анализа структурных свойств схемы. Так, система (рисунок 8.1) является статической, т.е. она имеет нулевой порядок астатизма, если выполняется условие

$$\lim_{p \rightarrow 0} W(p) = K < \infty, \quad (3.8)$$

где K – коэффициент усиления разомкнутой системы.

Для статической системы при постоянном входном воздействии $g(t) = A$ имеем:

$$\varepsilon = \lim_{p \rightarrow 0} p \cdot \frac{1}{1+W(p)} \cdot \frac{A}{P} = \frac{A}{1+K} \quad (3.9)$$

Последнее выражение означает, что постоянное входное воздействие обрабатывается с установившейся ошибкой, которую принято называть статической ошибкой. Для уменьшения статической ошибки необходимо увеличивать коэффициент усиления разомкнутой системы K .

При линейно нарастающем задающем входном воздействии $g(t) = V \cdot t$ имеем:

$$\varepsilon = \lim_{p \rightarrow 0} p \cdot \frac{1}{1+W(p)} \cdot \frac{V}{p^2} = \infty \quad (3.10)$$

Из полученного выражения следует, что линейно нарастающее воздействие обрабатывается статической системой с неограниченно растущей ошибкой.

Система автоматического управления (рис. 3.1) является астатической, если

$$\lim_{p \rightarrow 0} W(p) = \infty \quad (3.11)$$

и передаточная функция разомкнутой системы $W(p)$ может быть представлена в виде

$$W^*(p) = \frac{1}{p^r} \cdot W(p), \quad (3.12)$$

где $W(p)$ – передаточная функция статической системы, для которой выполняется условие (3.8); r – порядок астатизма системы.

Для астатической системы первого порядка при постоянном задающем воздействии $g(t) = A$ имеем:

$$\varepsilon = \lim_{p \rightarrow 0} p \cdot \frac{1}{1+\frac{W(p)}{P}} \cdot \frac{A}{P} = 0 \quad (3.13)$$

При линейно нарастающем задающем воздействии $g(t) = V \cdot t$

$$\varepsilon = \lim_{p \rightarrow 0} p \frac{1}{1 + \frac{W(p)}{P}} \cdot \frac{V}{P^2} = \lim_{p \rightarrow 0} \frac{P}{P + K} \cdot \frac{V}{P} = \frac{V}{K} \quad (3.14)$$

Установившиеся ошибки автоматического управления различного астатизма при типовых задающих воздействиях приведены в таблице 8.2.

Таблица 8.2 – Установившиеся ошибки систем

Порядок астатизма	$g(t) = A$	$g(t) = V \cdot t$	$g(t) = \frac{at^2}{2}$
0	$\frac{A}{1+K}$	∞	∞
1	0	$\frac{V}{K}$	∞
2	0	0	$\frac{a}{K}$

Аналогичным образом может быть введено понятие порядка астатизма по возмущающему воздействию. При этом следует отметить, что порядок астатизма по возмущающему воздействию не соответствует порядку астатизма по задающему воздействию.

В качестве примера рассмотрим задачу стабилизации величины $y(t)$ системы, представленной на рисунке 8.1.

На основе структурной схемы системы получаем при $g(t) = 0$:

$$y = W_0(p) \cdot R(p) \cdot (-y) + M(p)F, \quad (3.15)$$

где Y , F – соответственно изображения регулируемой величины и возмущающего воздействия.

Так как $W_0(p)R(p) = W(p)$, можно определить передаточную функцию замкнутой системы по возмущающему воздействию:

$$\Phi_f(p) = \frac{y}{F} = \frac{M(p)}{1 + W(p)} \quad (3.16)$$

При единичной отрицательной обратной связи и при $g(t) = 0$ имеем – $Y = E$, тогда передаточная функция замкнутой системы для ошибки по

возмущающему воздействию будет иметь тот же вид, что и для регулируемой величины, т. е.

$$\Phi_{\varepsilon f}(p) = \frac{E}{F} = -\Phi_f(p) \quad (3.17)$$

Таким образом, возмущающее воздействие f дает статическую ошибку

$$\varepsilon = \lim_{p \rightarrow 0} \left[-p \frac{M(p)}{1+W(p)} \cdot F \right] = -\frac{A}{1+K}, \quad (3.18)$$

где $M(p)=1$; $F = \frac{A}{p}$; $\lim_{p \rightarrow 0} W(p) = K$.

Порядок выполнения работы

1. Исследовать систему с астатизмом нулевого порядка. Структурная схема системы представлена на рисунке 8.2, где $R(p)=K$. Варианты передаточной функции $W_0(p)$ объекта управления и характеристики задающего воздействия $g(t)$ приведены в таблице 8.3.

1.1. Получить кривые переходного процесса для трех значений K ($K=1, 5, 10$) при подаче на вход системы сигнала $g(t)=A$ и определить предельные значения установившейся ошибки.

1.2. Получить кривые переходного процесса при подаче на вход системы линейно нарастающего воздействия $g(t)=V \cdot t$.

2. Исследовать систему с астатизмом первого порядка. В схеме (см. рис. 3.2) принять $R(p)=K/p$. Варианты передаточной функции $W_0(p)$ даны в таблице 8.4, а характеристики заданного воздействия $g(t)$ приведены в таблицах 8.3 и 8.4.

2.1. Получить кривые переходного процесса при подаче на вход системы задающего воздействия $g(t)=A$ (см. табл. 8.3).

2.2. Получить кривые переходного процесса при подаче на вход системы линейно нарастающего воздействия $g(t)=V \cdot t$ (см. табл.8.3). Определить предельные значения установившейся ошибки для различных значений коэффициента K ($K=1, 5, 10$).

2.3. Получить кривые переходного процесса при подаче на вход системы квадратично нарастающего воздействия $g(t)=a \cdot t^2/2$ (см. табл. 8.4).

3. Исследовать влияние внешнего возмущения.

3.1. В соответствии с вариантом (см. табл. 8.5 и рис. 8.3) собрать схему моделирования системы. При этом вид передаточной функции $W_0(p)$ взять из табл. 8.3.

3.2. Получить кривые переходного процесса и определить предельное значение установившейся ошибки ($g(t)=0, f(t)=1(t)$) и $R(p)=K, R(p)=K/p$.

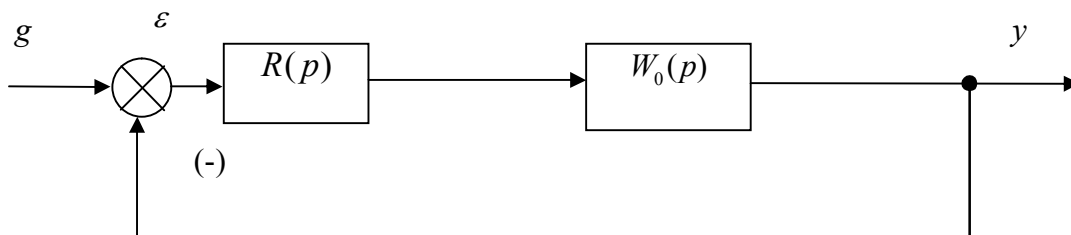


Рисунок 8.2 – Структурная схема системы

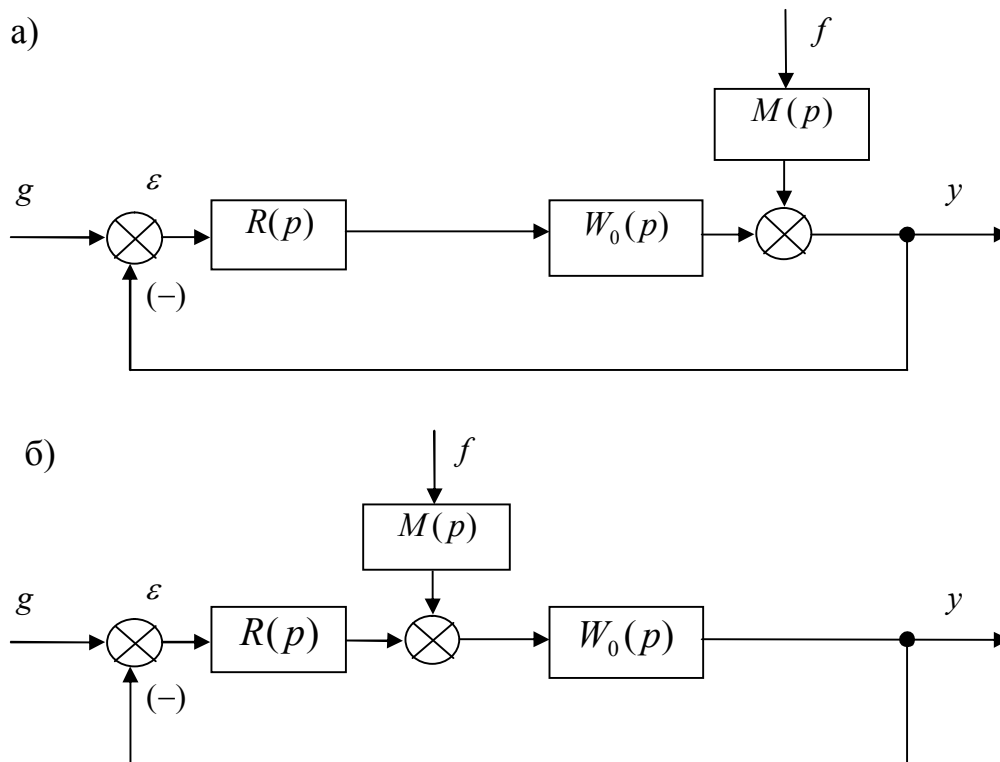


Рисунок 8.3 – Структурная схема системы при наличии возмущений

Таблица 8.3 – Варианты параметров систем с нулевым порядком астатизма

Вариант	$W_0(p)$	$g=A$	$g=V \cdot t$	Вариант	$W_0(p)$	$g=A$	$g=V \cdot t$
1	$\frac{2}{3p+1}$	1	0,5t	7	$\frac{1}{2p^2+3p+1}$	1	1,5t
2	$\frac{3}{2,5p+1}$	2	2t	8	$\frac{2}{0,5p^2+2p+1}$	1	2t
3	$\frac{1,5}{0,5p+1}$	2	4t	9	$\frac{2}{0,5p^2+p+2}$	2	2t
4	$\frac{1,5}{p^2+2p+1}$	1	t	10	$\frac{8}{0,5p^2+2p+8}$	2	t
5	$\frac{1}{p^2+p+2}$	2	2t	11	$\frac{1}{0,5p^2+p+1}$	2	2t
6	$\frac{5}{p^2+5p+6}$	1	t	12	$\frac{1}{0,1p^2+0,7p+1}$	4	2t

Таблица 8.4 – Варианты параметров астатических систем

Вариант	$W_0(p)$	$g = at^2/2$	Вариант	$W_0(p)$	$g = at^2/2$
1	$\frac{2}{3p+1}$	0,2t ²	7	$\frac{p+1}{2p^2+3p+1}$	0,25t ²
2	$\frac{3}{2,5p+1}$	0,5t ²	8	$\frac{p+2}{0,5p^2+2p+1}$	0,2t ²
3	$\frac{1,5}{0,5p}$	0,2t ²	9	$\frac{p+2}{0,5p^2+p+2}$	0,5t ²
4	$\frac{p+1,5}{p^2+2p+1}$	0,4t ²	10	$\frac{1,5p+8}{0,5p^2+2p+8}$	0,3t ²
5	$\frac{p+1}{p^2+p+2}$	0,3t ²	11	$\frac{p+1}{0,5p^2+p+1}$	0,45t ²
6	$\frac{p+5}{p^2+5p+6}$	0,45t ²	12	$\frac{p+1}{0,1p^2+0,7p+1}$	0,4t ²

Таблица 8.5 – Варианты параметров систем с возмущением

Вариант	1	2	3	4	5	6	7	8	9	10	11	12
Схема системы (№ рис.)	<i>a</i>	<i>б</i>	<i>a</i>	<i>б</i>	<i>a</i>	<i>б</i>	<i>A</i>	<i>б</i>	<i>a</i>	<i>б</i>	<i>a</i>	<i>б</i>
$M(p)$	0,5	0,5	1	2	0,5	2	1	2	2	1	0,5	1
f	2	2	-0,5	1	2	1	1	0,75	-0,75	2	-1	0,5

Пояснения по выполнению работы

1. Для «сборки» схемы моделирования системы с астатизмом нулевого порядка разместите на рабочем поле пакета SIMULINK усилитель, формирователь передаточных характеристик (Transfer function Block) и сумматор. Соедините компоненты как показано на структурной схеме (рис. 8.2). Установите параметры компонентов в соответствии с вариантом задания.

2. Для исследования точности статической системы при постоянном входном воздействии подключите к входу системы генератор сигналов из поля источников питания (Sources). Задайте нужное значение входного воздействия. Расположите осциллограф на рабочем поле пакета. На один вход осциллографа подайте задающее воздействие $g(t)$, на другой – ошибку $\varepsilon(t)$. Снимите зависимости $g(t)$ и $\varepsilon(t)$ для разных значений коэффициентов K .

3. Для исследования точности статической системы при линейно нарастающем входном воздействии включите последовательно с источником питания интегратор (Integrator). Задайте нужное значение скорости нарастания входного сигнала. Снимите зависимости $g(t)$ и $\varepsilon(t)$ для разных значений коэффициентов K .

4. При исследовании точности астатической системы первого порядка включите интегратор в структуру схемы модели последовательно с формирователем передаточных характеристик. Снимите зависимости $g(t)$ и $\varepsilon(t)$ при подаче на вход системы постоянного, линейно и квадратично нарастающих сигналов. Квадратично нарастающее входное воздействие можно получить, если включить последовательно с источником питания два интегратора.

5. Аналогично исследуйте влияние внешнего возмущающего воздействия на точность системы с астатизмом нулевого и первого порядков.

Вопросы для самоконтроля

1. Определить порядок астатизма системы автоматического управления (см. рисунок 8.2).

2. Чему равен коэффициент c_o в формуле (3.6) для системы с нулевым порядком астатизма?

3. Определить связь коэффициента усиления разомкнутой системы с предельным значением установившейся ошибки по задающему воздействию.

4. Можно ли компенсировать ошибку от возмущения, повысив порядок астатизма по задающему воздействию?

5. Определить порядок астатизма системы автоматического управления (см. рисунок 8.3).

Лабораторная работа №4.

ИССЛЕДОВАНИЕ УСТОЙЧИВОСТИ СИСТЕМ АВТОМАТИЧЕСКОГО УПРАВЛЕНИЯ

Цель работы: Экспериментальное построение областей устойчивости линейных систем автоматического управления и изучение влияния на устойчивость системы ее параметров.

Краткие теоретические сведения

Под устойчивостью САУ понимается способность системы возвращаться в заданное состояние или к заданному закону движения после отклонений, вызванными внешними возмущающими воздействиями.

Физической причиной неустойчивости замкнутых систем является инерционность их элементов, из-за чего воздействие обратной связи,

направленное на ликвидацию отклонения, запаздывает и поступает на вход объекта регулирования, когда отклонение уже изменилось. Этот процесс протекает либо в виде непрерывно возрастающего отклонения от заданного закона движения, либо в виде колебаний вокруг заданного значения выходной величины.

Устойчивость системы зависит от знака вещественных частей корней характеристического уравнения замкнутой системы:

$$\Phi(p) = \frac{M(p)}{D(p)},$$

$$D(p) = a_0 p^n + a_1 p^{n-1} + \dots + a_{n-1} p + a_n = 0.$$

Кроме этого корневого критерия устойчивости существуют косвенные критерии: алгебраические – Гаусса и Гурвица, частотные – Михайлова и Найквиста.

С повышением точности САУ, т.е. с увеличением коэффициента усиления, система становится менее устойчивой. Это объясняется тем, что с ростом коэффициента усиления на объект управления обратная связь действует сильнее. При этом увеличиваются отклонения под действием запаздывающего сигнала обратной связи.

Максимальный коэффициент, при котором система сохраняет устойчивость, называется критическим ($K_{кр}$).

Кроме коэффициента усиления, устойчивость зависит от инерционных свойств звеньев системы: постоянных времени и постоянных запаздывания. Поэтому устойчивость часто рассматривают как функцию двух или нескольких параметров. Обычно это – коэффициент усиления и постоянная времени одного из звеньев. На основании любого критерия устойчивости могут быть получены области устойчивости в плоскости двух параметров.

Под областью устойчивости в пространстве параметров понимается множество значений параметров, при которых система является асимптотически устойчивой.

Под областью неустойчивости, соответственно, понимается множество значений параметров, при которых система является неустойчивой. Области устойчивости и неустойчивости отделены друг от друга так называемыми границами устойчивости.

Граница устойчивости связывает выбранные параметры в предельном режиме перехода к неустойчивости, так что $K_{кр}=f(T)$.

Эта зависимость может быть получена расчетным путем на основе любого критерия устойчивости.

Например, по критерию устойчивости Михайлова система находится на границе устойчивости, если годограф

$$D(j\omega) = U_D(\omega) + jV_D(\omega)$$

проходит через начало координат.

Таким образом, уравнение границы устойчивости в пространстве варьируемых параметров K и T , согласно этому критерию примет вид:

$$U_D(K, T, \omega) = 0,$$

$$V_D(K, T, \omega) = 0.$$

Исключив из уравнения ω , можно вывести уравнение границы устойчивости, связывающее параметры T и $K_{кр}$.

Зависимость $K_{кр}=f(T)$ в данной работе определяется экспериментальным путем.

Порядок выполнения работы

1. Собрать схему модели системы в соответствии с вариантом задания.
2. Экспериментальным путем получить границу устойчивости системы $K_{кр}=f(T)$.
3. Выбрать точку на графике $K_{кр}=f(T)$. Построить годограф Михайлова для системы с выбранными параметрами.
4. Сравнить результаты эксперимента и расчета.

Пояснения по выполнению работы

1. Соберите схему моделирования системы, представленной на рисунке 4.1. Моделирование (сборку схемы и снятие переходных характеристик) проводите в последовательности, рассмотренной в лабораторной работе № 1 «Моделирование линейных систем автоматического управления».

2. Установите значение постоянной времени T_I в соответствии заданным вариантом (см. табл. 9.1).

3. Установите значение постоянной времени T равное $0,1c$.

4. Изменяя коэффициент усиления K , подберите такое его значение, при котором система находится на границе устойчивости. Тип устойчивости системы определяется по виду переходного процесса при нулевом входном воздействии $g(t)=0$ и при нулевом значении выходной переменной $y(0)=0$.

5. Для получения следующей точки границы устойчивости измените значение постоянной времени T . Количество точек, необходимых для построения границы устойчивости, должно быть не менее 10. Диапазон изменения постоянной времени T – от $0,1c$. до $5c$. Результаты эксперимента занесите в таблицу 9.2. Постройте график $K_{кр}=f(T)$.

6. Выберите точку на графике $K_{кр}=f(T)$. Для выбранных параметров системы постройте годограф Михайлова.

7. Сравните результаты расчета и эксперимента.

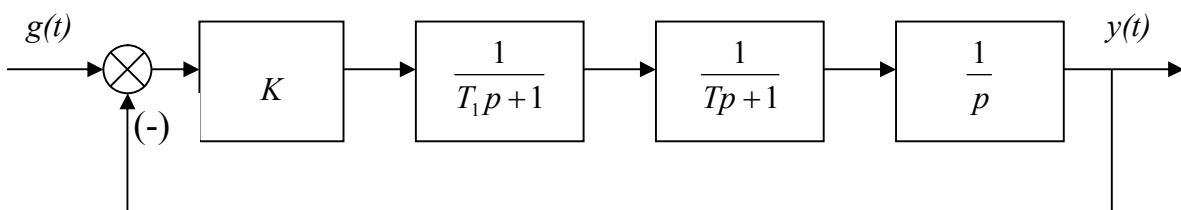


Рисунок. 9.1 – Структурная схема линейной системы автоматического управления

Таблица 9.1 – Варианты задания

№	1	2	3	4	5	6	7	8	9	10	11	12
п/п												
T, c	0,5	0,75	1,0	1,25	1,5	1,75	2,0	2,25	2,5	2,75	3,0	0,25

Таблица 9.2 – Граница устойчивости

T, c	0,1											5
$K_{кр}$												

Вопросы для самоконтроля

1. Сформулировать критерии устойчивости Гурвица, Михайлова, Найквиста, логарифмический критерий.
2. Как по логарифмическому критерию устойчивости определить $K_{кр}$ и $\omega_{кр}$?
3. Как построить $K_{кр}(T)$, используя критерий устойчивости Гурвица?
4. Сформулируйте корневой критерий устойчивости.

Лабораторная работа №5.

АНАЛИЗ ВЛИЯНИЯ ПОЛЮСОВ И НУЛЕЙ ПЕРЕДАТОЧНОЙ ФУНКЦИИ НА ДИНАМИЧЕСКИЕ СВОЙСТВА СИСТЕМЫ

Цель работы: Исследование связи динамических свойств системы управления с распределением полюсов и нулей ее передаточной функции.

Краткие теоретические сведения

В общем случае поведение системы описывается линейным дифференциальным уравнением n -го порядка

$$\frac{d^n y}{dt^n} + a_{n-1} \frac{d^{n-1} y}{dt^{n-1}} + \dots + a_1 \frac{dy}{dt} + a_0 y = \frac{d^m u}{dt^m} + b_{m-1} \frac{d^{m-1} u}{dt^{m-1}} + \dots + b_1 \frac{du}{dt} + b_0 u \quad (5.1)$$

где y – выходная переменная; u – управляющее воздействие.

Передаточная функция системы представляет собой дробно-рациональную функцию аргумента p полиномов $B(p)$ и $A(p)$.

$$\Phi(p) = \frac{B(p)}{A(p)} = \frac{p^m + b_{m-1}p^{m-1} + \dots + b_1p + b_0}{p^n + a_{n-1}p^{n-1} + \dots + a_1p + a_0} \quad (5.2)$$

где $a_0, \dots, a_{n-1}; b_0, \dots, b_{m-1}$ – постоянные параметры системы; $a_0=K$ – для астатических и $a_0=I+K$ – для статических систем.

Следует отметить, что в реальных системах всегда $m < n$.

Передаточную функцию можно представить в виде:

$$\Phi(p) = \frac{B(p)}{A(p)} = \frac{(p - \lambda_1)(p - \lambda_2) \dots (p - \lambda_m)}{(p - p_1)(p - p_2) \dots (p - p_n)} \quad (5.3)$$

где $\lambda_1, \lambda_2, \dots, \lambda_m$ – корни уравнения $B(p)=0$ или нули передаточной функции $\Phi(p)$; p_1, p_2, \dots, p_n – корни уравнения $A(p)=0$ или полюсы передаточной функции $\Phi(p)$.

Характер переходного процесса зависит как от знаменателя, так и от числителя передаточной функции системы. Полный анализ качества, поэтому может быть произведен лишь на основе анализа распределения не только полюсов передаточной функции (корней ее знаменателя), но и нулей (корней числителя). При анализе распределения полюсов и нулей необходимо определить:

- 1) связь параметров с распределением полюсов и нулей;
- 2) связь распределения полюсов и нулей с переходной функцией и ее основными показателями.

Если числитель передаточной функции не имеет нулей, т.е. равен постоянной величине, определяют только распределение полюсов. В этом случае передаточную функцию можно представить выражением

$$\Phi(p) = \frac{b}{p^n + a_{n-1}p^{n-1} + \dots + a_1p + a_0} \quad (5.4)$$

Полюсы передаточной функции (5.4) определяют вид переходной

функции $h(t)$ системы и, следовательно, такие динамические показатели, как время переходного процесса t_n и перерегулирование σ . При этом установившееся значение переходной функции рассчитывается по формуле $h(\infty) = b/a_0 = K$

Используя понятие *среднегеометрического корня*

$$\omega = \sqrt[n]{|p_1 p_2 \dots p_n|} = \sqrt[n]{a_0} \quad (5.5)$$

можно характеристический полином передаточной функции (5.4) представить в виде:

$$A(p) = p^n + \alpha_{n-1} \omega p^{n-1} + \dots + \alpha_1 \omega^{n-1} p + \omega^n \quad (5.6)$$

где $\alpha_1, \alpha_2, \dots, \alpha_{n-1}$ – коэффициенты, определяемые по формуле: $\alpha_i = \frac{a_i}{\omega^{n-i}}$

Среднегеометрический корень ω служит мерой быстродействия протекания переходных процессов. Если увеличить ω в 10 раз, то переходный процесс будет протекать в 10 раз быстрее. В связи с этим, при $\omega=1$ полином (5.6) следует рассматривать как *нормированный полином*, которому соответствует *нормированная переходная функция $h(\tau)$ и нормированное время переходного процесса τ* . Если качество переходного процесса, с точки зрения перерегулирования, является приемлемым, то требуемое время переходного процесса t_n можно обеспечить соответствующим выбором величины ω .

Для обеспечения требуемого значения перерегулирования необходимо задаться определенным распределением корней нормированного полинома, например, биномиальным распределением Ньютона или распределением Баттерворта.

Распределение Ньютона.

Выражение полинома Ньютона представим в виде:

$$A(p) = (p + \omega)^n = p^n + a_{n-1} \omega p^{n-1} + \dots + a_1 \omega^{n-1} p + \omega^n, \quad (5.7)$$

где $\alpha_{n-1}, \dots, \alpha_1$ – коэффициенты полинома.

Биномиальные значения коэффициентов могут быть рекомендованы для систем с передаточными функциями $\Phi(p)$, не имеющим нулей. Биномиальное распределение обеспечивает монотонный переходный процесс (т.е. $\sigma = 0\%$).

Таблица 10.1. Распределение Ньютона

Порядок полинома n	Выражение полинома
1	$P + \omega$
2	$P^2 + 2\omega P + \omega^2$
3	$P^3 + 3\omega P^2 + 3\omega^2 P + \omega^3$
4	$P^4 + 4\omega P^3 + 6\omega^2 P^2 + 4\omega^3 P + \omega^4$
5	$P^5 + 5\omega P^4 + 10\omega^2 P^3 + 10\omega^3 P^2 + 5\omega^4 P + \omega^5$
6	$P^6 + 6\omega P^5 + 15\omega^2 P^4 + 20\omega^3 P^3 + 15\omega^4 P^2 + 6\omega^5 P + \omega^6$

Распределение Баттерворта.

Корни полинома имеют вещественную часть $\text{Re}[p_i] \neq 0$ и равные модули $|p_i| = \omega$. Распределение Баттерворта обеспечивает системе управления хорошие динамические показатели с перерегулированием, не превышающим 15%. Коэффициенты нормированного полинома Баттерворта приведены в таблице 10.2.

Таблица 10.2 – Распределение Баттерворта

Порядок полинома n	Выражение полинома
1	$p + \omega$
2	$p^2 + 1,4\omega p + \omega^2$
3	$p^3 + 2\omega p^2 + 2\omega^2 p + \omega^3$
4	$p^4 + 2,61\omega p^3 + 3,4\omega^2 p^2 + 2,61\omega^3 p + \omega^4$
5	$p^5 + 3,24\omega p^4 + 5,24\omega^2 p^3 + 5,24\omega^3 p^2 + 3,24\omega^4 p + \omega^5$

В инженерных расчетах коэффициенты нормированных полиномов используются для расчета системы, описываемой уравнением

$$\frac{d^n y}{dt^n} + a_{n-1} \frac{d^{n-1} y}{dt^{n-1}} + \dots + a_1 \frac{dy}{dt} + a_0 y = bu \quad (5.8)$$

Коэффициенты уравнения (5.8) $a_i (i = 0, 1, \dots, n - 1)$ находятся по заданному значению времени переходного процесса t_n следующим образом:

- а) по нормированным переходным функциям определяется значение τ ,
- б) среднегеометрический корень ω рассчитывается по формуле $\omega = \tau / t_n$,
- в) коэффициенты α_i искомого полинома определяются выражением $a_i = \alpha_i \omega^{n-1}$, где α_i находятся по таблице 10.1 или 10.2 в зависимости от типа распределения корней характеристического уравнения.

Коэффициент числителя (5.4) определяется по заданной величине статического коэффициента K выражением $b = ka_0$.

В некоторых случаях возникает задача оценки быстродействия системы без построения ее переходной характеристики. Для этого может использоваться понятие степени устойчивости. Под степенью устойчивости η понимается расстояние от мнимой оси до ближайшего корня или до ближайшей пары комплексных сопряженных корней. Тогда время переходного процесса

$$t_n \approx \frac{1}{\eta} \ln \frac{1}{0,05} \approx \frac{3}{\eta} \quad (5.9)$$

Формула (5.9) обеспечивает приемлемую точность, если абсолютное значение вещественной части ближайшего к мнимой оси корня не менее чем на порядок меньше абсолютных значений вещественных частей всех остальных корней. При наличии нулей у передаточной функции (5.2), т.е. когда

$$B(p) = p^m + b_{m-1} p^{m-1} + \dots + b_1 p + b_0, \quad (5.10)$$

вывод о решающем влиянии степени устойчивости на длительность и характер переходной функции теряет силу. Нули передаточной функции при всех положительных b_0, b_1, \dots, b_{m-1} способствует перерегулированию. С другой

стороны выбором коэффициентов полинома (5.10) можно, например, уменьшить время переходного процесса, или обеспечить инвариантность системы к некоторым типам входных сигналов.

Порядок выполнения работы

1. Составить схему моделирования (табл. 10.3) системы, представленную уравнением (5.8).

2. Снять кривые переходных функций $h(\tau)$ нормированных полиномов Ньютона и Баттерворта (см. табл. 10.1 и 10.2).

3. Согласно заданным значениям t_n и K (см. табл. 10.4) рассчитать коэффициенты уравнения (5.8). Значения τ определяются по кривым переходных функций $h(\tau)$ нормированных полиномов Ньютона и Баттерворта.

4. Для каждого случая рассчитать корни характеристического уравнения и оценить время переходного процесса по формуле (5.9), затем снять кривые переходных процессов.

5. Составить схему моделирования, согласно уравнению (5.1).

6. Используя набор параметров b_0, b_1, \dots, b_{m-1} приведенных в табл. 10.3 дополнить передаточную функцию (5.4) с коэффициентами, рассчитанными в п.3 для полинома Ньютона, до вида передаточной функции (5.2).

7. Снять кривые переходных процессов.

8. Сделать сравнительный анализ результатов моделирования.

Таблица 10.3 – Варианты параметров моделей

Вариант	1	2	3	4	5	6	7	8	9	10	11	12
a_0	9	5	5	8	7	15	7	2	1	25	30	0,12
a_1	6	4	4	6	5	5	3	0,5	0,5	1	0,8	1
a_2	3	3	2	2	10	2	2	0,2	0,2	0,2	0,2	0,2
a_3	1	1	1	1	3	1	1	0,1	0,1	0,1	0,1	0,1
b_0	12	2,5	7,5	12	10	15	10	4	2	25	30	0,1
b_1	2	2	0	1	3	0,5	6	2	2	2	3	2
b_2	0,1	3	5	10	1,5	1	3	1	1	1	1	1
b_3	0,05	1	1	1	0,1	0,5	1	0,5	0,5	0,5	0,5	0,5

Таблица 10.3

Вариант	1	2	3	4	5	6	7	8	9	10	11	12
K	10	20	25	100	50	40	30	48	60	70	20	15
t_n	0,1	0,2	0,5	0,1	0,4	0,1	0,2	0,5	0,1	0,4	0,5	0,6

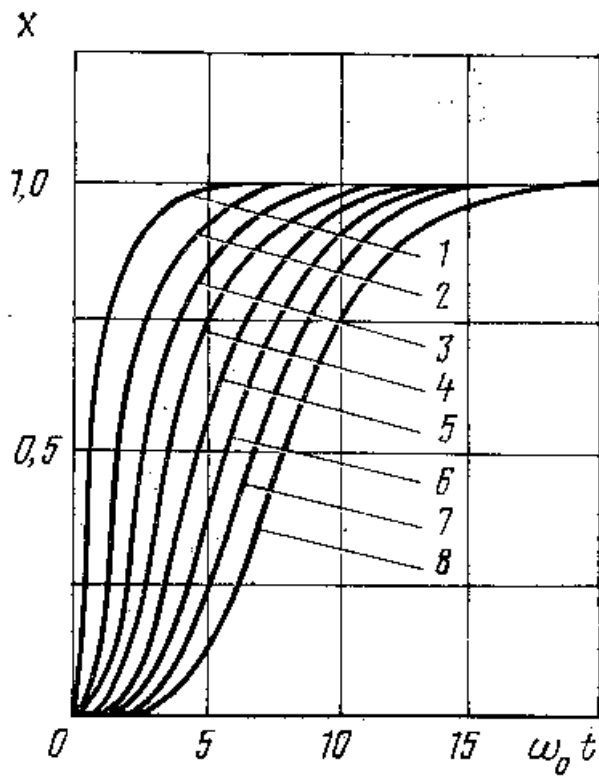


Рисунок 10.1 – Переходные характеристики нормированных полиномов Ньютона

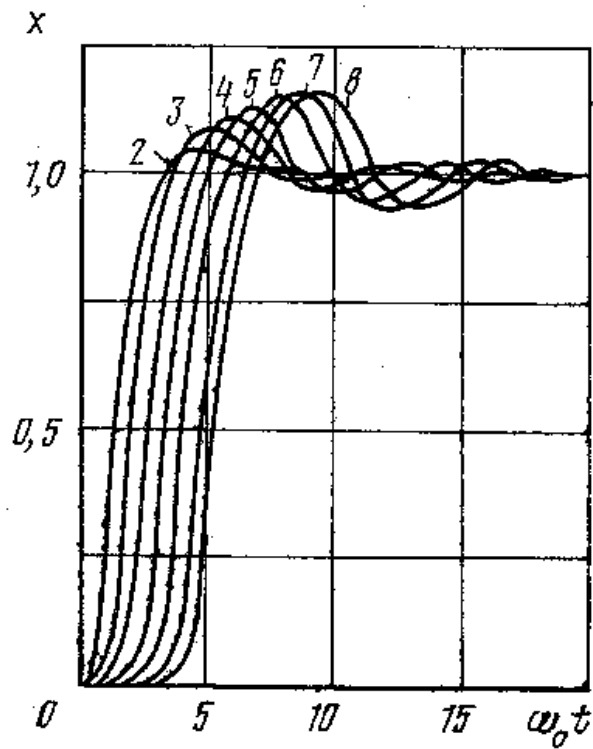


Рисунок 10.2 – Переходные характеристики нормированных полиномов Баттерворта

Литература

1. **Черных И. В.** SIMULINK: среда создания инженерных приложений. Под общ. ред. к.т.н. В. Г. Потемкина. – М.: ДИАЛОГ-МИФИ, 2003. – 496 с.
2. **Поршнева С. В.** MATLAB 7. Основы работы и программирования. Учебник. – М.: ООО "Бином-Пресс", 2006. – 320 с.
3. **Мартынов Н.Н., Иванов А.П.** MATLAB 5.x Вычисления. Визуализация. Программирование. – М.: Кудиц-Образ, 2000.- 332с.
4. **Ануфриев И.Е., Смирнов А.Б., Смирнова Е.Н.** MATLAB 7. - СПб: БХВ-Петербург, 2005. - 1104 с.
5. **Чен К.** MATLAB в математических исследованиях: учебник / К. Чен, Джиблин П. – Москва: Мир, 2001. – 346 с.
6. **Дьяконов В. П.** MATLAB 6.5/7.0 + Simulink 5/6. Основы применения. Библиотека профессионала. – Москва.: «СОЛОН-Пресс», 2005. – 800 с.

Для заметок

Ахметзянов И. З., Каримов В. С.

МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ В СРЕДЕ SIMULINK

Лабораторный практикум

Подписано в печать _____. Бумага офсетная. Формат 60x84 1/16

Печать ризографическая. Гарнитура Times New Roman Cug.

Усл. печ. л. 3,0 Уч.- изд. л. 3,0. Тираж 100 экз. Заказ № _____

Отпечатано с готового оригинал-макета в лаборатории оперативной полиграфии Набережночелнинского института(филиала) Казанского (Приволжского) федерального университета

423823, Республика Татарстан г. Набережные Челны, пр. Мира 68/19