

Министерство образования и науки Украины
Севастопольский национальный технический университет



МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ

по выполнению лабораторной работы
«Применение программы «SIMULINK» для функционального
моделирования радиоэлектронной аппаратуры»
по дисциплине «Основы компьютерного
проектирования и моделирования РЭА»
для студентов дневной и заочной форм обучения
направления 0907 — радиотехника

Севастополь

2006

Методические рекомендации по выполнению лабораторной работы «Применение программы «SIMULINK» для функционального моделирования радиоэлектронной аппаратуры» по дисциплине «Основы компьютерного проектирования и моделирования РЭА» для студентов дневной и заочной форм обучения направления 0907 — радиотехника / Сост. В. В. Вертегел — Севастополь: Изд-во СевНТУ, 2006. — 31с.

Целью рекомендаций является оказание помощи студентам в изучение принципов функционального моделирования РЭА с использованием пакета моделирования SIMULINK.

Методические рекомендации рассмотрены и утверждены на заседании кафедры радиотехники (протокол № _____ от _____ 2006 г.)

Допущено учебно-методическим центром СевНТУ в качестве методических указаний.

Рецензент: к.т.н., доцент кафедры радиотехники Бадалов А.Г.

СОДЕРЖАНИЕ

Введение	4
1. Краткое описание программы	5
1.1. Назначение окон программы <i>MATLAB</i>	5
1.2. Назначение и основные возможности программы <i>Simulink</i>	6
1.3. Запуск программы <i>Simulink</i>	7
1.4. Обзорщик разделов библиотеки <i>Simulink</i>	8
1.5. Создание модели.....	10
1.6. Окно модели.....	12
1.7. Основные приемы подготовки и редактирования модели	14
1.8. Установка параметров расчета и его выполнение	18
1.8.1. Установка параметров расчета модели	19
1.8.2. Параметры вывода результатов моделирования	20
1.8.3. Установка параметров обмена с рабочей областью <i>MATLAB</i>	20
1.8.4. Установка параметров диагностирования модели.....	21
1.9. Выполнение расчета.....	22
1.10. Завершение работы.....	22
2. Пример моделирования RC-автогенератора.....	23
2.1. Краткие теоретические сведения	23
2.2. Пример построения модели и её анализа.....	26
3. Индивидуальное задание	28
4. Содержание отчета	28
5. Контрольные вопросы.....	28
Приложение А. Состав библиотеки блоков <i>Simulink</i>	29

ВВЕДЕНИЕ

Среди большого числа пакетов прикладных программ система *MATLAB* занимает особое место. Первоначально ориентированная на исследовательские проекты, система в последние годы стала рабочим инструментом не только учёных, но также инженеров-разработчиков и студентов. В сообществе радиоинженеров, *MATLAB* получил необычайное распространение, и, по сути, стала средством международного общения. Особенно широко, эффективно и эффективно система *MATLAB* применяется в области обработки сигналов, которая затрагивает информатику и связь, радиолокацию и радионавигацию, радиовещание и телевидение, медицинское приборостроение и измерительную технику, автомобильную и бытовую электронику и многое другое.

MATLAB — одна из старейших, тщательно проработанных и проверенных временем систем автоматизации математических расчетов, построенная на применении матричных операций. Это нашло отражение в названии системы — *MATrix LABoratory* — матричная лаборатория. Однако синтаксис языка программирования системы продуман настолько тщательно, что это почти не ощущается теми пользователями, которых не интересуют непосредственно матричные вычисления.

В обширном и постоянно пополняемом комплексе команд, функций и пакетов расширения (*toolboxes*) системы *MATLAB* содержатся специальные средства для электротехнических и радиотехнических расчетов (операции с комплексными числами, матрицами, векторами и полиномами, обработка данных, анализ сигналов и цифровая фильтрация), обработки изображений, а также средства, относящиеся к другим новым направлениям науки и техники.

Важными достоинствами системы являются ее открытость и расширяемость. Большинство команд и функций системы реализованы в виде текстовых *m*-файлов (с расширением *.m*) и файлов на языке Си, причем все файлы доступны для модификации. Пользователю дана возможность создавать не только отдельные файлы, но и библиотеки файлов для реализации специфических задач.

Возможности *MATLAB* весьма обширны, а по скорости выполнения задач система нередко превосходит своих конкурентов (*Mathcad*, *Mathematica*, *Maple*). Она применима для расчетов практически в любой области науки и техники. Этому способствует не только расширенный набор матричных и иных операций и функций, но и наличие разнообразных пакетов расширения, среди которых *Simulink* — предназначенный для решения задач функционального моделирования динамических систем и устройств различного назначения.

Simulink — сопутствующая *MATLAB* интерактивная графическая программа, которая позволяет моделировать динамические системы на уровне структурных и функциональных схем. Это позволяет эффективно использовать *Simulink* на этапе структурного и функционального моделирования радиоэлектронной аппаратуры.

1. КРАТКОЕ ОПИСАНИЕ ПРОГРАММЫ

1.1. Назначение окон программы *MATLAB*

После запуска *MATLAB* на экране открывается главное окно (рис.1.1), составными частями (фреймами) которого являются:

- командное окно (**Command Window**),
- окно быстрого запуска (**Launch Pad**),
- окно рабочего пространства (**Workspace**).
- окно истории команд (**Command History**),
- окно текущей директории (**Current Directory**).

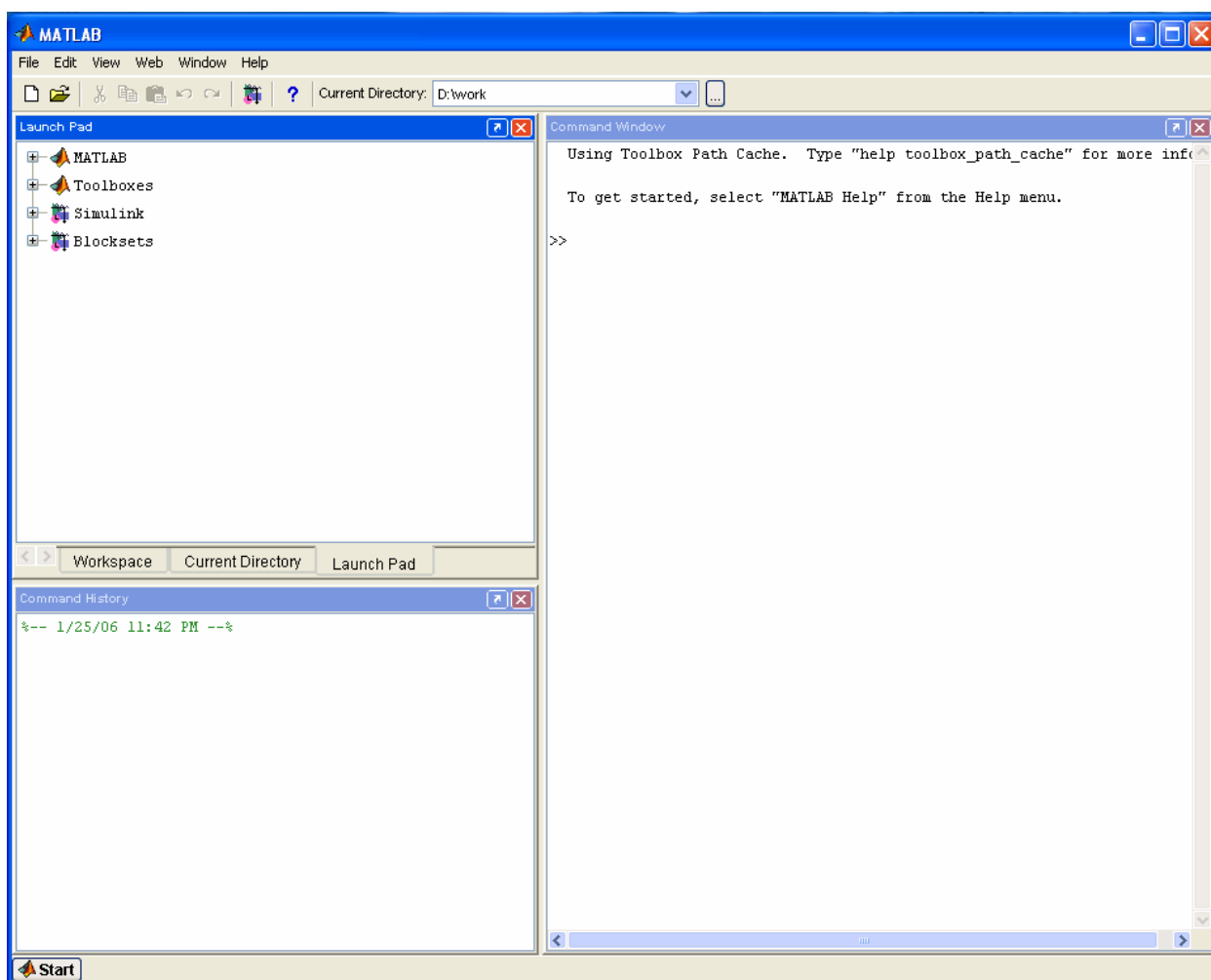


Рис.1.1 — Главное окно программы *MATLAB*

Среди перечисленных окон главную роль играет командное окно, позволяющее производить вычисления в диалоговом режиме:

- в строке с приглашением в виде знака `>>` набирается имя команды или вычисляемое математическое выражение, после чего нажимается клавиша ввода (**Enter**);
- результат вычислений отображается в командном окне.

Вводимую в строке команду можно редактировать обычными для

MS DOS и *Windows* способами: удалять символы слева и справа от курсора клавишами **Backup** и **Delete**, перемещаться по строке влево и вправо клавишами со стрелками $\leftarrow$ и $\rightarrow$. Однако есть особенность: нажатием клавиш со стрелками ($\uparrow$ и $\downarrow$) можно вызвать из памяти отдававшиеся ранее команды.

Остальные окна играют служебную роль. Среди них окно рабочего пространства (**Workspace**). Оно позволяет контролировать тип и размерность переменных, участвующих в вычислениях.

Окно истории команд (**Command History**) представляет собой дневник, в котором в хронологическом порядке хранятся все отдававшиеся когда-либо команды. Очистку окон — командного, рабочего пространства и истории проще всего производить из пункта меню главного окна: **Edit/Clear**

Окно текущей директории (**Current Directory**) отображает папки и файлы, находящиеся в текущей, т.е. рабочей директории.

Окно быстрого запуска (**Launch Pad**) позволяет находить и запускать основные программные модули *MATLAB* и пакеты расширения, среди которых *Simulink* и *Blocksets* — дополнения к *Simulink*, содержащих библиотеки блоков для специализированных приложений, таких как связь, обработка сигналов, энергетические системы и пр..

1.2. Назначение и основные возможности программы *Simulink*

Программа *Simulink* является приложением к пакету *MATLAB*. При моделировании с использованием *Simulink* реализуется принцип визуального программирования, в соответствии с которым пользователь на экране из библиотеки стандартных блоков создает модель устройства и осуществляет расчеты. При этом, в отличие от классических способов моделирования, пользователю не нужно досконально изучать язык программирования и численные методы математики, а достаточно общих знаний, требующихся для работы на компьютере и, естественно, знаний той предметной области, в которой он работает.

Simulink позволяет имитировать работу реальных систем и устройств на основе их моделей, составленных из функциональных блоков. Программа имеет обширную и расширяемую пользователями библиотеку блоков и средства задания и изменения их параметров. Построение модели устройства или системы сводится к перемещению с помощью мыши необходимых блоков из библиотек *Simulink* в окно создаваемой модели и соединению этих блоков между собой. Работая с программой *Simulink*, можно создавать модели линейных и нелинейных, аналоговых, дискретных и смешанных (аналогово-дискретных) цепей и систем, изменять параметры блоков непосредственно во время процесса моделирования и сразу же наблюдать реакцию моделируемой системы.

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. Результаты моделирования могут быть представлены в виде графиков или таблиц.

Simulink является достаточно самостоятельным инструментом *MATLAB* и при работе с ним совсем не требуется знать сам *MATLAB* и остальные его приложения. С другой стороны доступ к функциям *MATLAB* и другим его инструментам остается открытым и их можно использовать в *Simulink*. Часть входящих в состав *MATLAB* пакетов имеет инструменты, встраиваемые в *Simulink* (например, *LTI-Viewer* приложения Control System Toolbox — пакета для разработки систем управления). Имеются также дополнительные библиотеки блоков для разных областей применения (например, *Digital Signal Processing Blockset* — набор блоков для разработки устройств цифровой обработки сигналов).

Simulink имеет более 100 встроенных блоков, реализующих большой набор функциональных преобразований, необходимых для моделирования разнообразных устройств и систем. Блоки сгруппированы в библиотеки в соответствии с их назначением: источники сигнала, приемники, дискретные, непрерывные, нелинейные, математические операции, функции и таблицы, сигналы и системы. В дополнение к обширному набору встроенных блоков *Simulink* имеет расширяемую библиотеку блоков благодаря функции создания пользовательских блоков и библиотек. Например, можно создать блоки для моделирования приёмников и передатчиков (преобразователи, фильтры, детекторы, генераторы, модуляторы и пр.). Преимущество *Simulink* заключается также в том, что он позволяет пополнять библиотеки блоков с помощью подпрограмм написанных как на языке *MATLAB*, так и на языках *C++*, *Fortran* и *Ada*.

1.3. Запуск программы *Simulink*

Запуск программы *Simulink* можно выполнить из главного окна *MATLAB* одним из трех способов:

1. Нажать пиктограмму  на панели инструментов главного окна *MATLAB*.
2. В командном окне *MATLAB* напечатать “**simulink**” и нажать клавишу **Enter** на клавиатуре.
3. Выполнить команду **Open...** в меню **File** и открыть файл модели (**mdl** - файл).

Последний вариант удобно использовать для запуска уже готовой и отлаженной модели, когда требуется лишь провести расчеты и не нужно добавлять новые блоки в модель. Использование первого и второго способов приводит к открытию окна обозревателя разделов библиотеки (**Simulink Library Browser**) (рис. 1.2). Это окно представляет собой некоторое хранилище (библиотеку) блоков функциональных преобразований сигналов, из которых можно построить и испытать функциональную схему устройства, а на основе функциональных моделей устройств создать и исследовать структурные схемы радиосистем различного назначения.

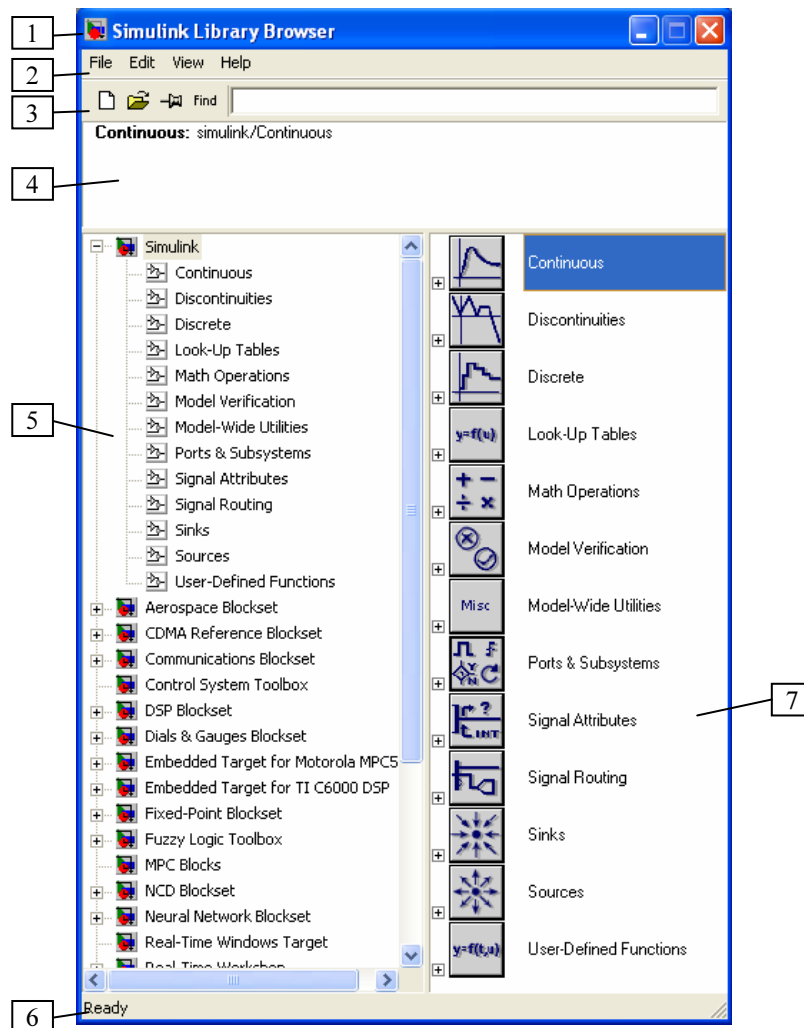


Рис. 1.2 — Окно обозревателя разделов библиотеки *Simulink*

1.4. Обозреватель разделов библиотеки *Simulink*

Окно обозревателя библиотеки блоков содержит следующие элементы:

1. Заголовок, с названием окна — **Simulink Library Browser**.
2. Меню, с командами **File**, **Edit**, **View**, **Help**.
3. Панель инструментов, с ярлыками наиболее часто используемых команд.
4. Окно комментария для вывода поясняющего сообщения о выбранном блоке.
5. Список разделов библиотеки, реализованный в виде дерева.
6. Строка состояния, содержащая подсказку по выполняемому действию.
7. Окно содержимого раздела библиотеки (список вложенных разделов библиотеки или блоков)

На рис. 1.2 выделена основная библиотека *Simulink* (в левой части окна) и показаны ее разделы (в правой части окна).

Блоки библиотеки *Simulink* сгруппированы по следующим группам:

- **Continuous** — аналоговые блоки;
- **Discontinuous** — блоки квантования;
- **Discrete** — дискретные блоки;
- **Look-Up Tables** — блоки таблиц;
- **Math Operations** — блоки математических операций;
- **Model Verification** — блоки верификации модели;
- **Model-Wide Utilities** — информационные и вспомогательные блоки;
- **Ports & Subsystem** — блоки портов и подсистем;
- **Signal Attributes** — блоки задания параметров сигналов;
- **Signal Routing** — блоки маршрутизации сигналов;
- **Sinks** — приемники сигналов;
- **Sources** — источники сигналов;
- **User-Defined Function** — функции заданные пользователем.

В приложении А приведён состав наиболее часто используемых блоков.

Список разделов библиотеки *Simulink* представлен в виде дерева, и правила работы с ним являются общими для списков такого вида:

- пиктограмма свернутого узла дерева содержит символ "+", а пиктограмма развернутого содержит символ "-";
- для того чтобы развернуть или свернуть узел дерева, достаточно щелкнуть на его пиктограмме левой клавишей мыши.

При выборе соответствующего раздела библиотеки в правой части окна отображается его содержимое (рис. 1.3).

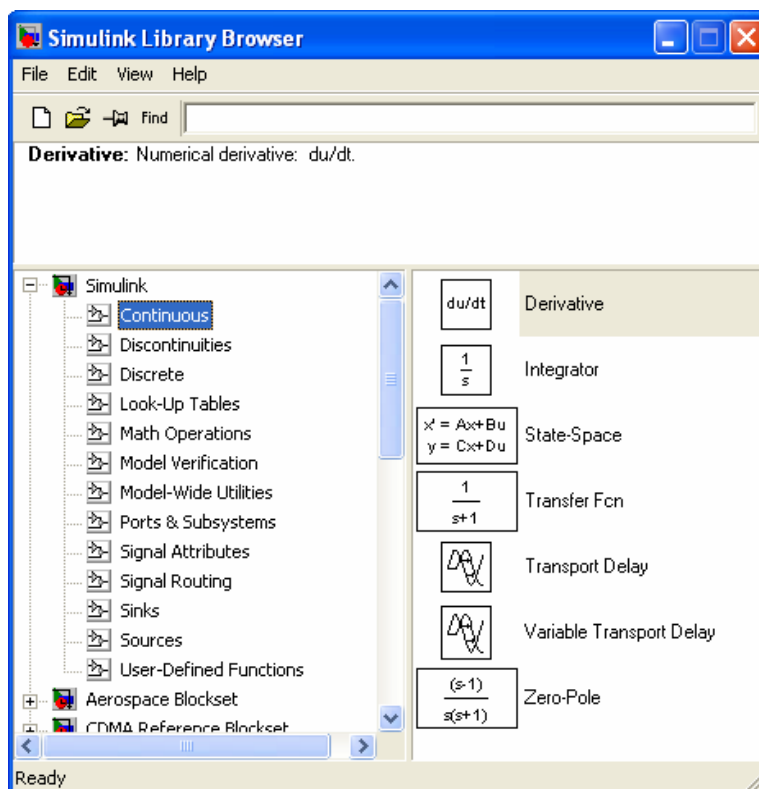


Рис. 1.3 — Окно обозревателя с набором блоков раздела библиотеки

Для работы с окном используются команды собранные в меню. Меню обозревателя библиотек содержит следующие пункты:

- **File (Файл)** — Работа с файлами библиотек.
- **Edit (Редактирование)** — Добавление блоков и их поиск (по названию).
- **View (Вид)** — Управление показом элементов интерфейса.
- **Help (Справка)** — Вывод окна справки по обозревателю библиотек.

Для работы с обозревателем можно также использовать кнопки на панели инструментов (рис. 1.4).

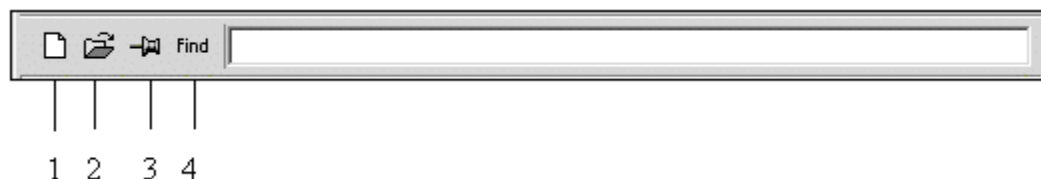


Рис. 1.4 — Панель инструментов обозревателя разделов библиотек

Кнопки панели инструментов имеют следующее назначение:

1. Создать новую модель (открыть новое окно модели).
2. Открыть одну из существующих моделей.
3. Изменить свойства окна обозревателя. Данная кнопка позволяет установить режим отображения окна обозревателя “поверх всех окон”. Повторное нажатие отменяет такой режим.
4. Поиск блока по названию (по первым символам названия). После того как блок будет найден, в окне обозревателя откроется соответствующий раздел библиотеки, а блок будет выделен. Если же блок с таким названием отсутствует, то в окне комментария будет выведено сообщение **Not found <имя блока>** (Блок не найден).

1.5. Создание 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).

2. Расположить блоки в окне модели. Для этого необходимо открыть соответствующий раздел библиотеки (Например, **Sources — Источники**). Далее, указав курсором на требуемый блок и нажав на левую клавишу “мыши” — “перетащить” блок в созданное окно. *Клавишу мыши нужно держать нажатой.* На рис. 1.5 показано окно модели, содержащее блоки.

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

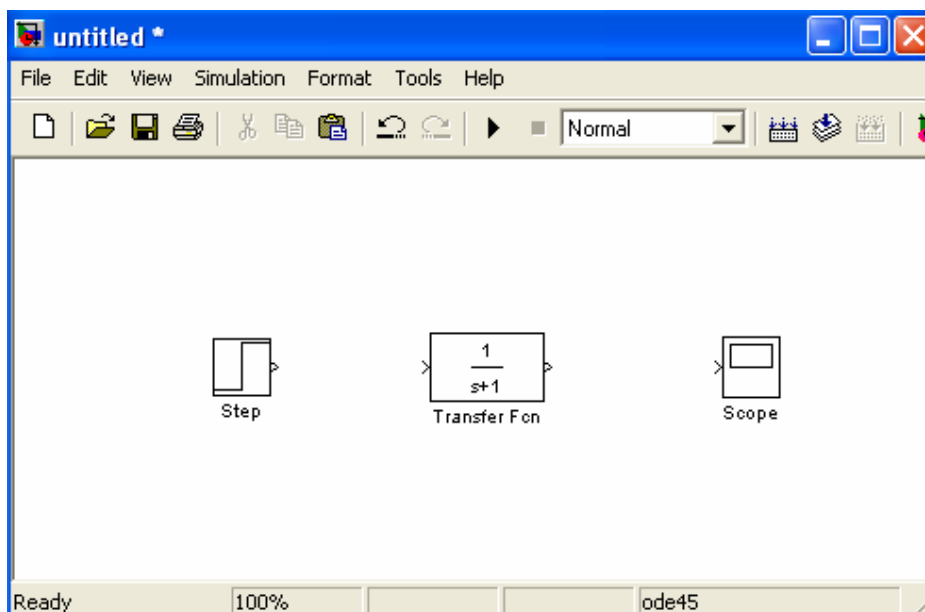


Рис. 1.5 — Окно модели, содержащее блоки

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 рис. 1.6 в качестве примера показан блок, моделирующий передаточную функцию и окно редактирования параметров данного блока.

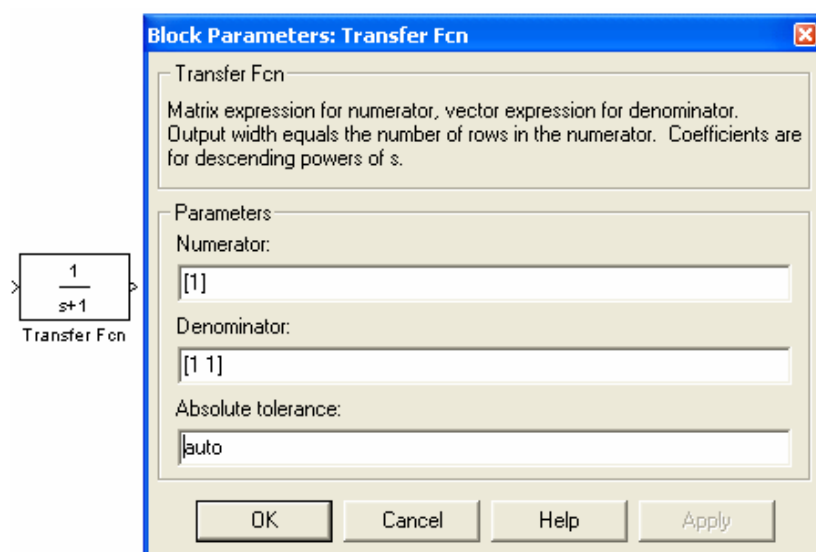


Рис. 1.6 — Блок, моделирующий передаточную функцию и окно редактирования параметров блока

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. Для удаления линии требуется выбрать линию, а затем нажать клавишу **Delete** на клавиатуре. Схема модели, в которой выполнены соединения между блоками, показана на рис. 1.7.

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 символов, 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*.

1.6. Окно модели

Окно модели содержит следующие элементы (см. рис. 1.7):

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.

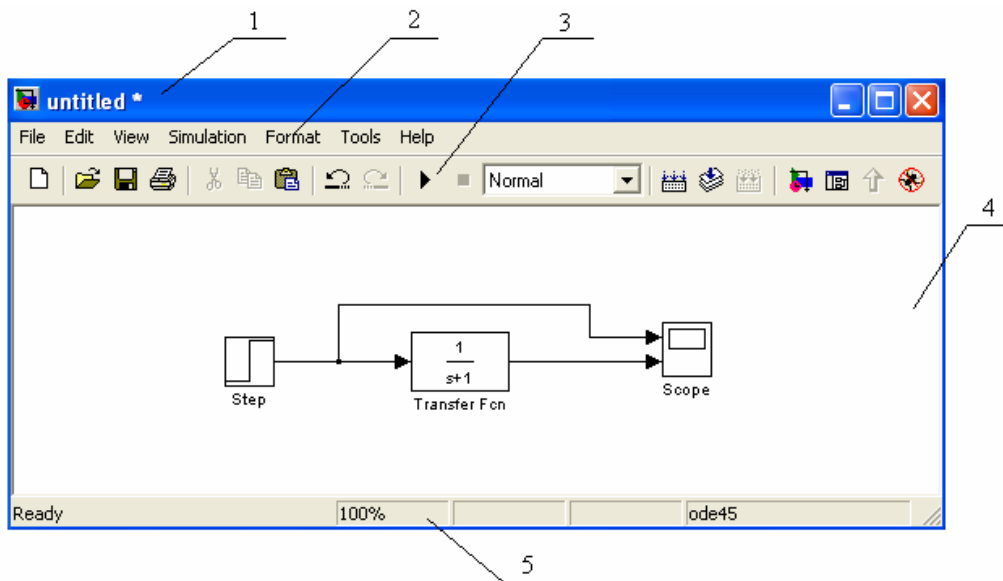


Рис.1.7 — Окно модели

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 управление 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 (рис. 1.8.).

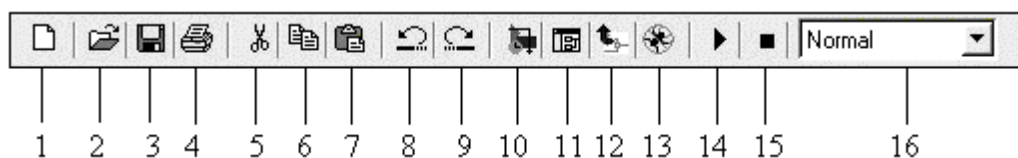


Рис. 1.8 — Панель инструментов окна модели

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 , и ей соответствует уже команда **Pause** (Приостановить мо-

делирование); 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).

1.7. Основные приемы подготовки и редактирования модели

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 мыши. Произойдет выделение объекта. Об этом будут свидетельствовать маркеры по углам объекта. 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**. После выделения объекта его можно копировать или перемещать в буфер промежуточного хранения, извлекать из буфера, а также удалять, используя стандартные приемы работы в *Windows*-программах.

Копирование и перемещение объектов в буфер промежуточного хранения

Для копирования объекта в буфер его необходимо предварительно выделить, а затем выполнить команду **Edit/Copy** или воспользоваться пиктограммой  на панели инструментов.

Для вырезания объекта в буфер его необходимо предварительно выделить, а затем выполнить команду **Edit/Cut** или воспользоваться пиктограммой  на панели инструментов. При выполнении данных операций следует иметь в виду, что объекты помещаются в собственный буфер *MATLAB* и недоступны из других приложений. Использование команды **Edit/Copy model to Clipboard** позволяет поместить графическое изображение модели в буфер **Windows** и, соответственно, делает его доступным для остальных программ.

Копирование можно выполнить и таким образом: нажать правую клавишу мыши, и не отпуская ее, переместить объект. При этом будет создана копия объекта, которую можно переместить в необходимое место.

Вставка объектов из буфера промежуточного хранения

Для вставки объекта из буфера необходимо предварительно указать место вставки, щелкнув левой клавишей мыши в предполагаемом месте вставки, а затем выполнить команду **Edit/Paste** или воспользоваться инструментом  на панели инструментов.

Удаление объектов

Для удаления объекта его необходимо предварительно выделить, а затем выполнить команду **Edit/Clear** или воспользоваться клавишей **Delete** на клавиатуре. Следует учесть, что команда **Clear** удаляет блок без помещения его в буфер обмена. Однако эту операцию можно отменить командой меню **File/Undo**.

Соединение блоков

Для соединения блоков необходимо сначала установить курсор мыши на выходной порт одного из блоков. Курсор при этом превратится в большой крест из тонких линий (рис. 1.9). Держа нажатой левую кнопку мыши, нужно переместить курсор ко входному порту нужного блока. Курсор мыши примет вид креста из тонких сдвоенных линий (рис. 1.10). После создания линии необходимо отпустить левую клавишу мыши. Свидетельством того, что соединение создано, будет жирная стрелка у входного порта блока. Выделение линии производится точно также как и выделение блока – одинарным щелчком левой клавиши мыши. Черные маркеры, расположенные в узлах соединительной линии будут говорить о том, что линия выделена.

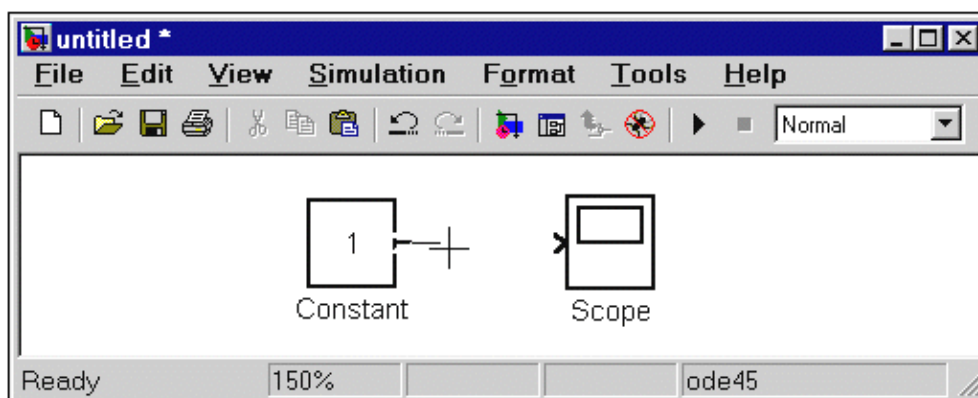


Рис. 1.9 — Начало создания соединения

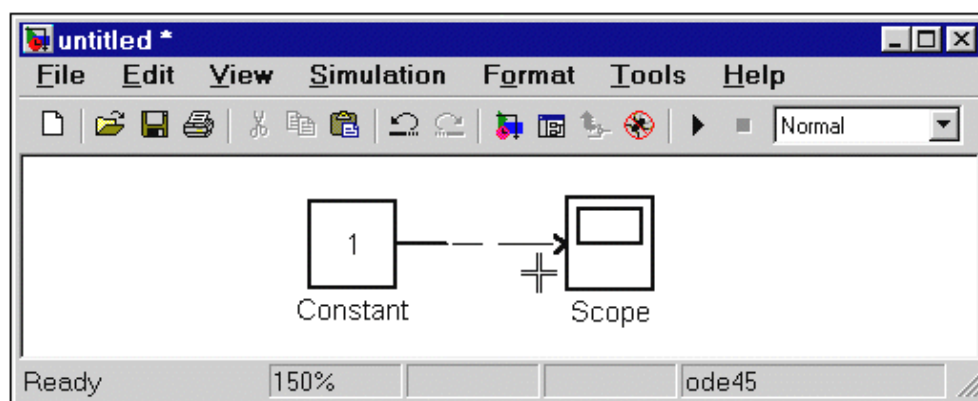


Рис. 1.10 — Завершение создания соединения

Создание петли линии соединения выполняется также как перемещение блока. Линия соединения выделяется, и затем нужная часть линии перемещается. Рисунок 1.11 поясняет этот процесс.

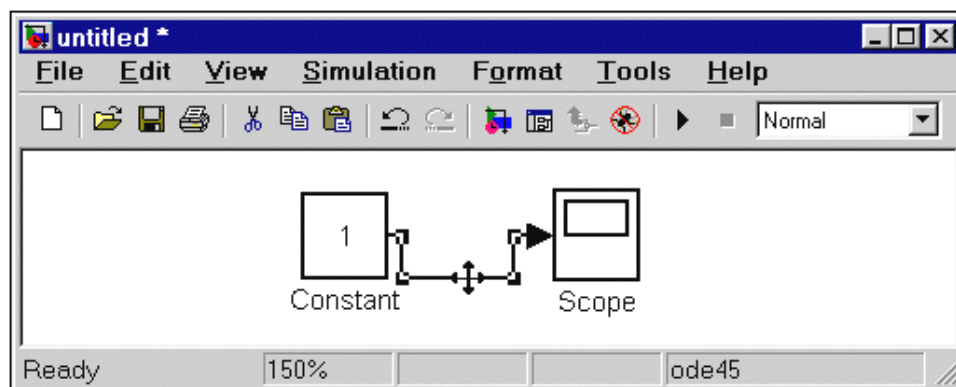


Рис. 1.11 — Создание петли в соединительной линии

Удаление соединений выполняется также как и любых других объектов (см. п. 1.7.4.).

Изменение размеров блоков

Для изменения размера блока он выделяется, после чего курсор мыши надо установить на один из маркеров по углам блока. После превращения курсора в двустороннюю стрелку, необходимо нажать левую клавишу мыши и растянуть (или сжать) изображения блока. На рис. 1.12 показан этот процесс. Размеры надписей блока при этом не изменяются.

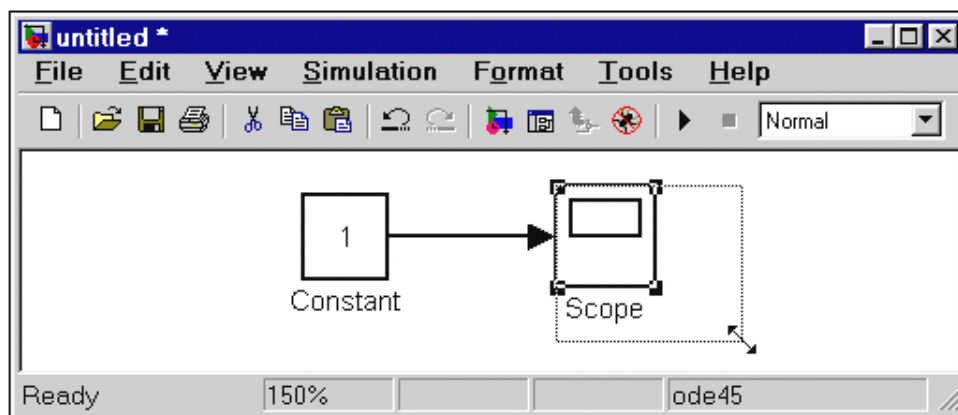


Рис. 1.12 — Изменение размера блока

Перемещение блоков

Любой блок модели можно переместить, выделив его, и передвинув, держа нажатой левую клавишу мыши. Если ко входам и выходам блока подведены соединительные линии, то они не разрываются, а лишь сокращаются или увеличиваются в длине. В соединение можно также вставить блок, имеющий один вход и один выход. Для этого его нужно расположить в требуемом месте соединительной линии.

Использование команд Undo и Redo

В процессе освоения программы пользователь может совершать действия кажущиеся ему необратимыми (например, случайное удаление части модели, копирование и т.д.). В этом случае следует воспользоваться командой **Undo** — отмена последней операции. Команду можно вызвать с помощью кнопки  в панели инструментов окна модели или из меню **Edit**. Для восстановления отмененной операции служит команда **Redo** (пиктограмма .

Форматирования 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.
 - **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 не скалярных 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.

1.8. Установка параметров расчета и его выполнение

Перед выполнением расчетов необходимо предварительно задать параметры расчета. Задание параметров расчета выполняется в панели управления меню **Simulation/ Simulation Parameters**. Вид панели управления приведен на рис. 1.13.

Окно настройки параметров расчета имеет 5 вкладок:

- **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.
- **Real-Time Workshop (Отладчик реального времени)** — Установка параметров отладки модели.

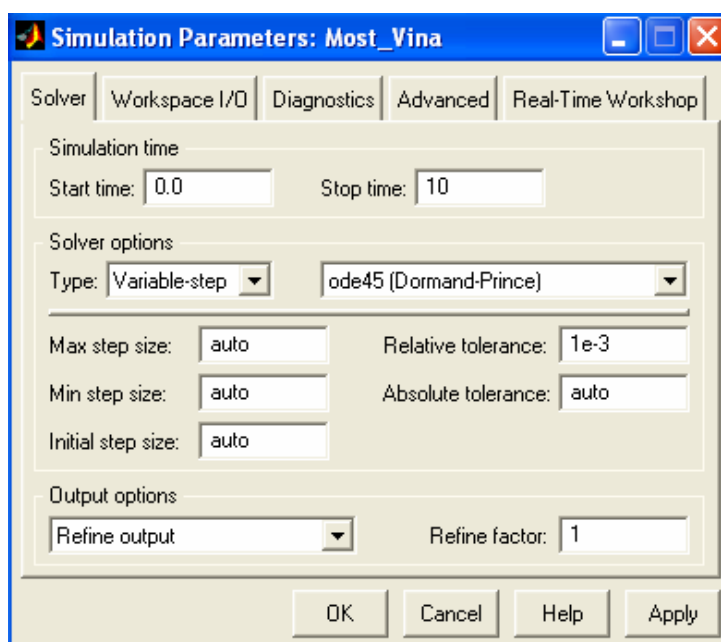


Рис. 1.13 — Панель управления

Установка параметров расчета модели выполняется с помощью элементов управления, размещенных на вкладке **Solver**. Эти элементы разделены на три группы (рис. 1.13): **Simulation time** (Интервал моделирования или, иными словами, время расчета), **Solver options** (Параметры расчета), **Output options** (Параметры вывода).

1.8.1. Установка параметров расчета модели

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.

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.

Ниже двух раскрывающихся списков **Type** находится область, содержание которой меняется в зависимости от выбранного способа изменения модель-

ного времени. При выборе **Fixed-step** в данной области появляется текстовое поле **Fixed-step size** (величина фиксированного шага) позволяющее указывать величину шага моделирования. Величина шага моделирования по умолчанию устанавливается системой автоматически (**auto**).

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**.

1.8.2. Параметры вывода результатов моделирования

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 равно 1. Это означает, что регистрация производится с шагом $\Delta t = 1$ (то есть для каждого значения модельного времени:).
- **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.
- **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).

1.8.3. Установка параметров обмена с рабочей областью *MATLAB*

Элементы, позволяющие управлять вводом и выводом в рабочую область *MATLAB* промежуточных данных и результатов моделирования, расположены на вкладке **Workspace I/O** (рис. 1.14).

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**).

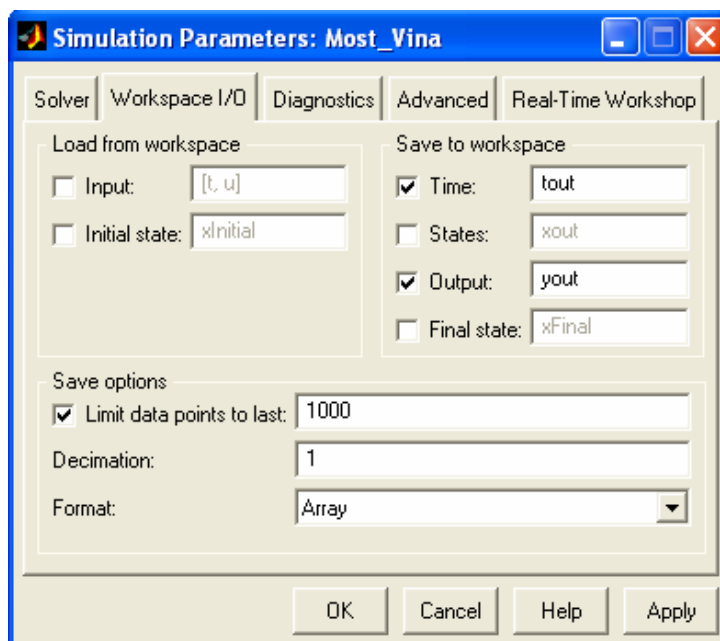


Рис. 1.14 — Вкладка **Workspace I/O** диалогового окна установки параметров моделирования

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. 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”).

1.8.4. Установка параметров диагностирования модели

Вкладка **Diagnostics** (рис. 1.15) 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.

Выборанный вид реакции отображается в списке рядом с наименованием события.

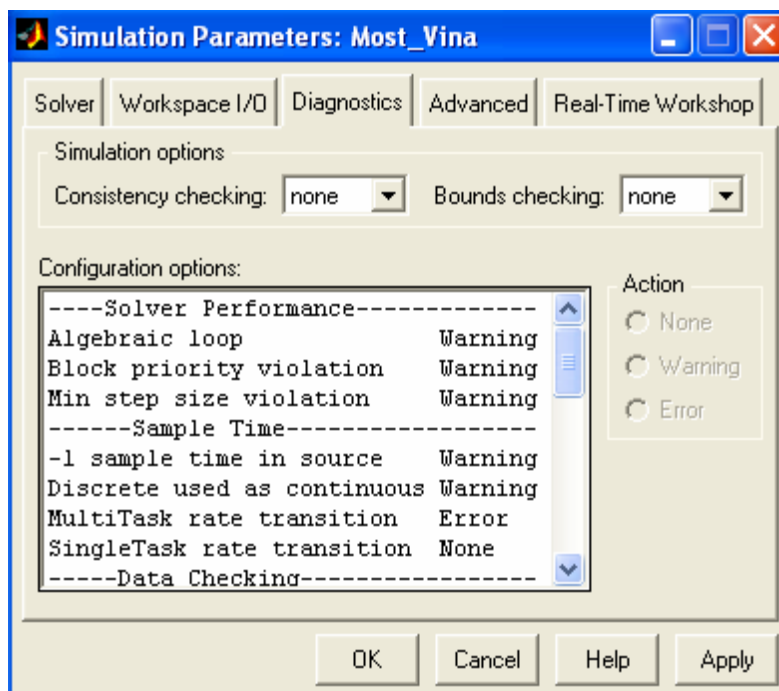


Рис. 1.15 — Вкладка **Diagnostics** окна установки параметров моделирования

1.9. Выполнение расчета

Запуск расчета выполняется с помощью выбора пункта меню **Simulation/Start**, или кнопки ► на панели инструментов окна модели. Процесс расчета можно завершить досрочно, выбрав пункт меню **Simulation/Stop** или кнопку ■. Расчет также можно остановить (**Simulation/Pause**) и затем продолжить (**Simulation/Continue**).

1.10. Завершение работы

Для завершения работы необходимо сохранить модель в файле, закрыть окно модели, окно обозревателя библиотек, а также основное окно пакета *MATLAB*.

2. ПРИМЕР МОДЕЛИРОВАНИЯ RC-AВТОГЕНЕРАТОРА

2.1. Краткие теоретические сведения

Рассмотрим процесс функционального моделирования на примере RC-автогенератора. Напомним, что схема такого генератора состоит из усилителя и цепи частотно-зависимой обратной связи. В качестве такой цепи обычно используют мост Вина (рис. 2.1.).

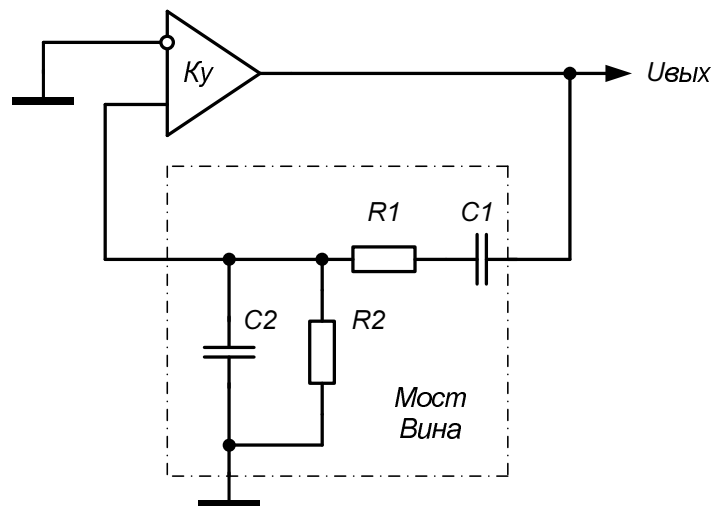


Рис. 2.1 — Упрощенная схема RC-автогенератора

При $R1 = R2 = R$, $C1 = C2 = C$ передаточная функция моста Вина может быть записана следующим образом:

$$K(p) = \frac{\tau p}{\tau^2 p^2 + 3\tau p + 1}; \quad (2.1)$$

где $\tau = RC$, $p = j\omega$.

Самовозбуждение автогенератора (возникновение незатухающих колебаний на частоте $\omega_r = 1/RC = 1/\tau$) наступает при выполнении условий:

$$\begin{cases} K_y \cdot K_{oc} > 1, \\ \varphi_y + \varphi_{oc} = 2\pi n, \end{cases} \quad (2.2)$$

где K_y — коэффициент усиления усилителя; K_{oc} — коэффициент передачи цепи обратной связи; φ_y — сдвиг фаз, вносимый усилителем; φ_{oc} — сдвиг фаз, вносимый цепью обратной связи; $n = 0, \pm 1, \pm 2 \dots$

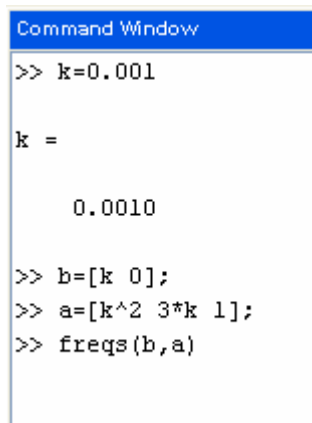
Зададимся величиной $R1 = R2 = 1$ кОм, $C1 = C2 = 1$ мкФ, тогда $\tau = 0,001$, $\omega_r = 1 \cdot 10^3$ (рад/с).

Если $\varphi_y = 0$ (для неинвертирующего усилителя на рис. 2.1), то второе условие (2.2) выполняется в том случае, когда $\varphi_{oc} = 0, \pm 2\pi, \pm 4\pi \dots$

Для определения коэффициента передачи цепи обратной связи K_{oc} на частоте ω_r и коэффициента усиления усилителя K_y , необходимых для выполнения первого условия (2.2), воспользуемся функцией программы *MATLAB*.

Построение графиков модуля и аргумента комплексного коэффициента передачи моста Вина выполним при помощи функции **freqs(b,a)**. Здесь **b** и **a** — векторы коэффициентов полиномов, соответственно числителя и знаменателя передаточной функции (2.1). Коэффициенты следуют в порядке убывания степеней, заканчивая постоянным слагаемым.

Предварительно в командном окне программы *MATLAB* после приглашения в виде знака **>>** введём значение **k = 0.001**, затем нажмем клавишу ввода (**Enter**). (Буквой **k** обозначим τ , поскольку синтаксис *MATLAB* не допускает использования греческих букв.) После ввода коэффициентов **b** и **a** командное окно программы *MATLAB* для расчета передаточной функции моста Вина будет иметь вид, показанный на рис. 2.2.



```

Command Window
>> k=0.001

k =

    0.0010

>> b=[k 0];
>> a=[k^2 3*k 1];
>> freqs(b,a)
  
```

Рис. 2.2 — Вид командного окна для расчета передаточной функции моста Вина

После нажатия клавиши **Enter** функция **freqs(b,a)** по умолчанию строит графики АЧХ в логарифмическом масштабе, а ФЧХ в линейном (ось частот имеет логарифмический масштаб). В результате на экране появляется окно **Figure No. 1**, показанное на рис. 2.3.

Для определения коэффициента передачи моста Вина на частоте $\omega_r = 1 \cdot 10^3$ (рад/с) изменим масштаб оси "Y" графика АЧХ с логарифмического на линейный. Для этого в строке меню окна **Figure No. 1** выберем пункт **Edit/Axes Properties...** и в открывшемся окне **Property Editor-Axes** установим для оси "Y" линейный масштаб. В результате окно с графиками АЧХ и ФЧХ будет иметь вид, показанный на рис. 2.4.

По графикам определим, что значение модуля комплексного коэффициента передачи моста Вина $K_{oc} = 0,33$, а сдвиг фаз, вносимый мостом, $\varphi_{oc} = 0$ на частоте $\omega_r = 1 \cdot 10^3$ (рад/с). Таким образом, для выполнения первого условия (2) необходимо, чтобы коэффициент усиления усилителя K_y был больше 3.

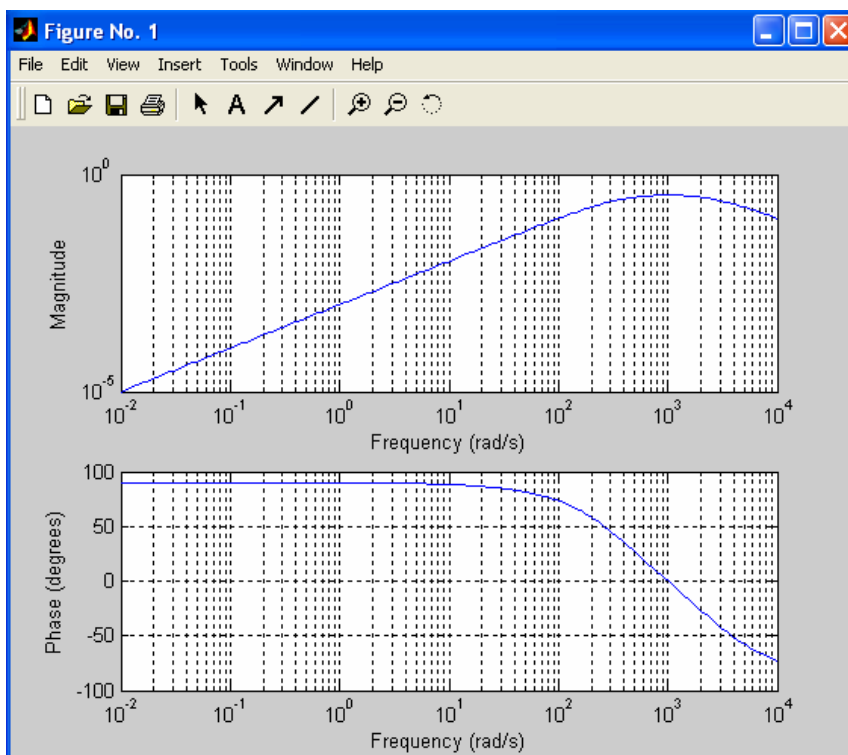


Рис. 2.3 — Графики АЧХ и ФЧХ моста Вина

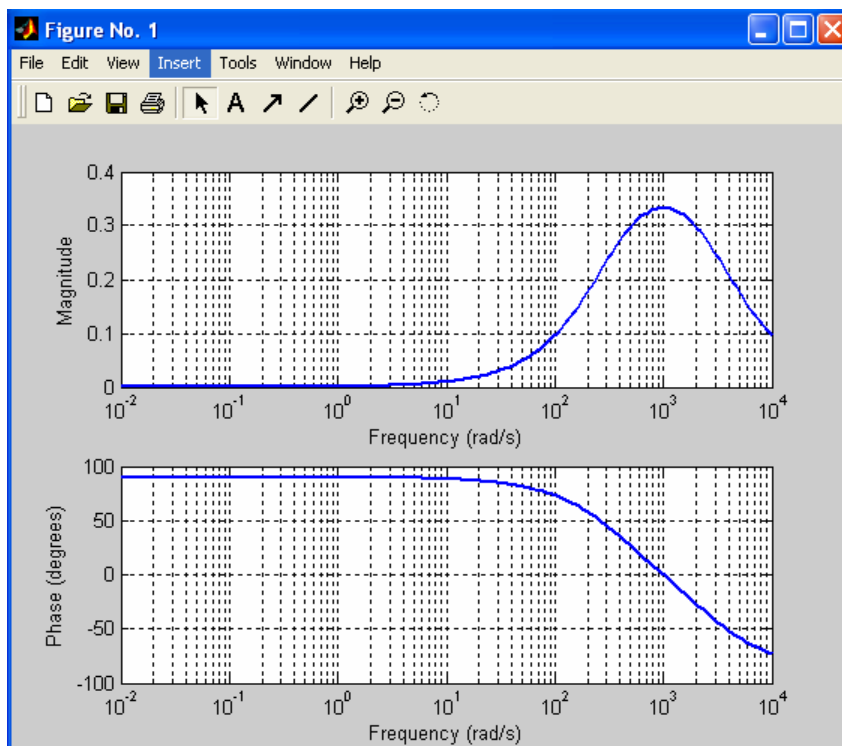


Рис. 2.4 — Графики АЧХ и ФЧХ в линейном масштабе

Для исследования процессов в RC-автогенераторе разработаем его функциональную модель. В качестве элементов модели будем использовать стандартные блоки библиотеки *Simulink*.

2.2. Пример построения модели и её анализа

Для построения функциональной модели генератора необходимо:

1. Создать новый файл модели с помощью команды **File/New/Model**, или используя кнопку  на панели инструментов окна обозревателя разделов библиотеки **Simulink Library Browser**.

2. Поместим в окно модели блок генератора ступенчатого сигнала (**Step**), находящегося в группе блоков **Sources**. Отметим, что, кроме непосредственного просмотра содержимого библиотек, любой блок может быть найден по имени (если оно известно), введённому в текстовое поле, расположенное в правой верхней части **Simulink Library Browser**.

Кроме источника сигнала нам потребуются следующие блоки, которые также следует поместить в окно модели:

- сумматор (**Sum**) и блок усиления (**Gain**), находящихся в группе блоков математических операций (**Math Operations**);
- блок передаточной функции (**Transfer Fcn**), находящегося в группе аналоговых блоков (**Continuous**);
- осциллограф (**Scope**), находящегося в группе приёмников сигналов (**Sinks**).

После соединения входов и выходов блоков схема модели будет иметь вид, показанный на рис. 2.5.

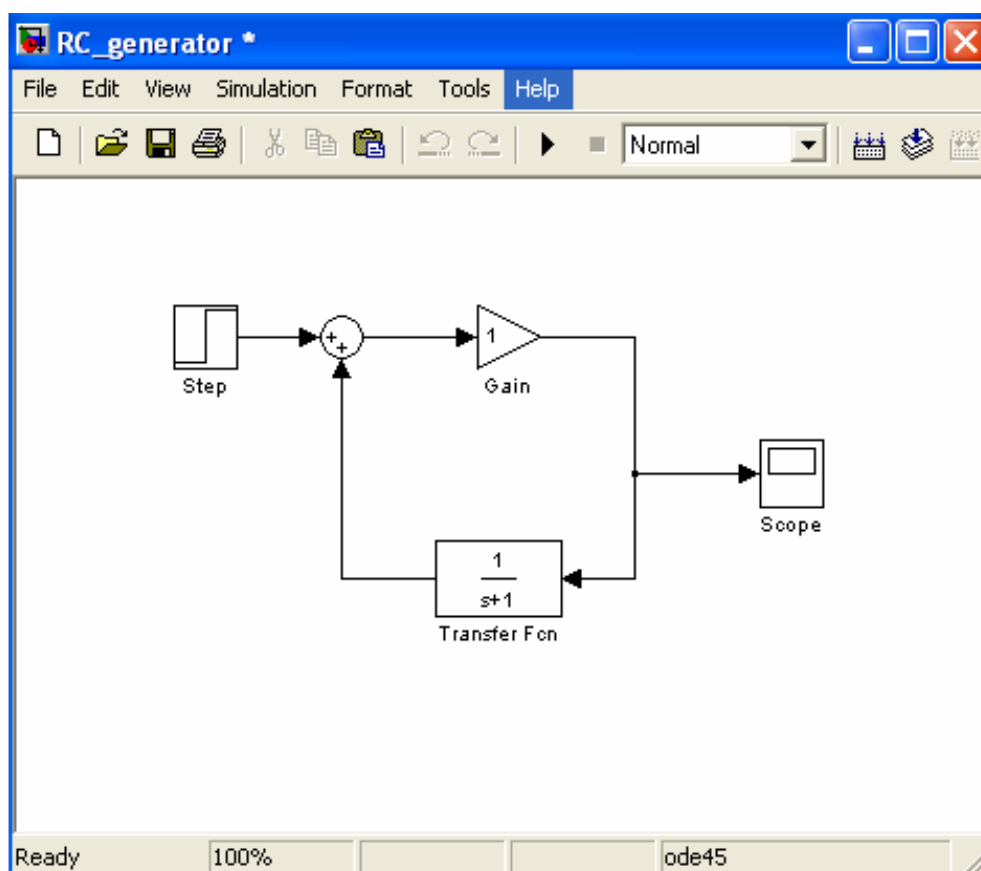


Рис. 2.5 — Модель RC-автогенератора

3. Установим параметры блоков. Коэффициент усиления усилителя установим равным 2,5. Для этого двойным щелчком левой клавиши мыши по блоку **Gain** откроем окно редактирования его параметров (**Block Parameters: Gain**). Вместо 1 введем значение коэффициента усиления $K_y = 2,5$.

Двойным щелчком левой клавиши мыши по блоку передаточной функции **Transfer Fcn** откроем окно редактирования его параметров. Установим в качестве параметров числителя передаточной функции вектор **b**, а знаменателя вектор **a** (рис. 2.6).

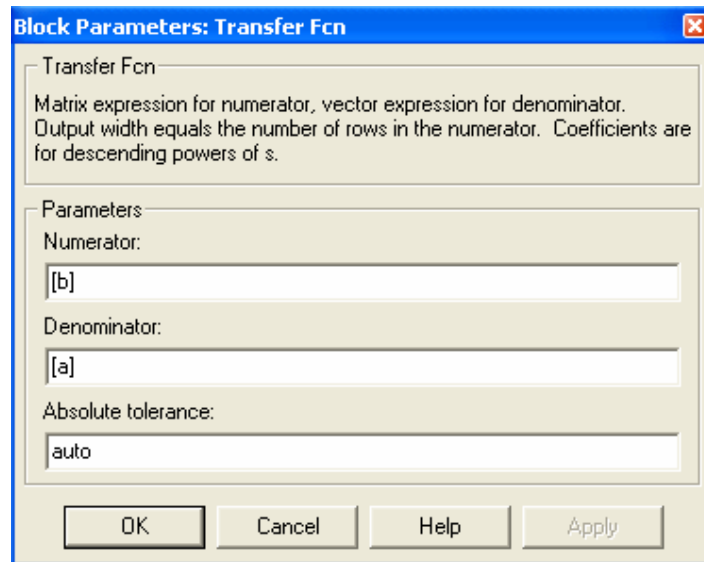


Рис. 2.6 — Параметры блока передаточной функции

4. Используя пункт меню окна модели **Simulation/ Simulation Parameters** установим соединяющие параметры времени расчета (**Simulation time**): **Start time = 0; Stop time = 0.1**. Прочие параметры расчета оставим без изменений.

5. Двойным щелчком левой клавиши мыши по блоку генератора ступенчатого сигнала (**Step**) откроем окно редактирования его параметров. Установим параметр **Step time = 0.01**. Прочие параметры расчета оставим без изменений.

6. Двойным щелчком левой клавиши мыши по осциллографу (**Scope**) откроем его окно. В открывшемся окне **Scope** в панели инструментов нажмем пиктограмму  (**Parameters**). Установим параметр **Time range = 0.1**. Прочие параметры расчета оставим без изменений.

7. Выбрав пункт меню **Simulation/Start**, или кнопки  на панели инструментов окна модели запустим процесс моделирования.

В результате выполнения расчета на экране осциллографа появится выходной сигнал, являющийся реакцией на входное воздействие в виде ступенчатого сигнала в момент времени $t = 0.01$ с.

8. Установим коэффициент усиления блока **Gain** $K_y = 3$ и опять запустим процесс моделирования. В этом случае в автогенераторе установится стационарный режим колебаний с частотой $\omega_T = 1 \cdot 10^3$ (рад/с).

3. ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ

1. Разработать и исследовать функциональную модель амплитудного модулятора.
2. На основе модели амплитудного модулятора разработать исследовать модель функционального генератора, формирующего амплитудно-модулированные колебания с коэффициентом модуляции $M = 0.5$. Амплитуда несущих колебаний равна 1 В. Частота несущих колебаний должна быть равна $n + 1$ кГц, а частота модулирующих колебаний $(n + 1)/10$ кГц, где n — последняя цифра номера зачетной книжки.
3. Дополнить модель функционального генератора блоками, обеспечивающими формирование сигнала у которого амплитуда несущих колебаний равна 1 В, а частота за $2(n + 1)$ мс изменяется от $n + 1$ до $2(n + 1)$ кГц. За это же время модулирующее колебание в виде прямоугольного импульса с частотой $(n + 1)/10$ кГц и скважностью $N = 2$ увеличивается по амплитуде от 0.1 до 0.5 В.

4. СОДЕРЖАНИЕ ОТЧЕТА

1. Цель работы.
2. Схема модели RC-автогенератора и результаты его исследований.
3. Схема модели амплитудного модулятора и результаты его исследований.
4. Схема модели функционального генератора и осциллограммы его выходных сигналов.
5. Выводы по работе.

5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Назначение окон программы *MATLAB*.
2. Назначение и основные возможности программы *Simulink*.
3. Назначение и состав окна обозревателя разделов библиотеки (Simulink Library Browser).
4. Последовательность создания модели устройства.
5. Основные приемы подготовки и редактирования модели
6. Назначение и основные параметры используемых для создания модели блоков.
7. Назначение и основные параметры расчета модели.
8. Параметры вывода результатов моделирования.
9. Параметров обмена с рабочей областью *MATLAB*.
10. Установка параметров диагностирования модели.

Приложение А

Состав библиотеки блоков *Simulink*

1. Sources — источники сигналов

- 1.1 Источник постоянного сигнала Constant
- 1.2. Источник синусоидального сигнала Sine Wave
- 1.3. Источник линейно изменяющегося воздействия Ramp
- 1.4. Генератор ступенчатого сигнала Step
- 1.5. Генератор сигналов Signal Generator
- 1.6. Источник случайного сигнала с равномерным распределением Uniform Random Number
- 1.7. Источник случайного сигнала с нормальным распределением Random Number
- 1.8. Источник импульсного сигнала Pulse Generator
- 1.9. Генератор линейно-изменяющейся частоты Chirp Generator
- 1.10. Генератор белого шума Band-Limited White Noise
- 1.11. Источник временного сигнала Clock
- 1.12. Цифровой источник времени Digital Clock
- 1.13. Блок считывания данных из файла From File
- 1.14. Блок считывания данных из рабочего пространства From Workspace
- 1.15. Блок сигнала нулевого уровня Ground
- 1.16. Блок периодического сигнала Repeating Sequence
- 1.17. Блок входного порта Inport

2. Sinks — приемники сигналов

- 2.1. Осциллограф Scope
- 2.2. Осциллограф Floating Scope
- 2.3. Графопостроитель XY Graph
- 2.4. Цифровой дисплей Display
- 2.5. Блок остановки моделирования Stop Simulation
- 2.6. Блок сохранения данных в файле To File
- 2.7. Блок сохранения данных в рабочей области To Workspace
- 2.8. Концевой приемник Terminator
- 2.9. Блок выходного порта Outport

3. Continuous — аналоговые блоки

- 3.1. Блок вычисления производной Derivative
- 3.2. Интегрирующий блок Integrator
- 3.3. Блок Memory
- 3.4. Блок фиксированной задержки сигнала Transport Delay
- 3.5. Блок управляемой задержки сигнала Variable Transport Delay
- 3.6 Блок передаточной функции Transfer Fcn
- 3.7. Блок передаточной функции Zero-Pole
- 3.8. Блок модели динамического объекта State-Space

4. Discrete — дискретные блоки

- 4.1. Блок единичной дискретной задержки Unit Delay

- 4.2. Блок экстраполятора нулевого порядка Zero-Order Hold
- 4.3. Блок экстраполятора первого порядка First-Order Hold
- 4.4. Блок дискретного интегратора Discrete-Time Integrator.
- 4.5. Дискретная передаточная функция Discrete Transfer Fcn
- 4.6. Блок дискретной передаточной функции Discrete Zero-Pole
- 4.7. Блок дискретного фильтра Discrete Filter
- 4.8. Блок модели динамического объекта Discrete State-Space

5. Discontinuous — блоки квантования

- 5.1. Блок ограничения Saturation
- 5.2. Блок с зоной нечувствительности Dead Zone
- 5.3. Релейный блок Relay
- 5.4. Блок ограничения скорости изменения сигнала Rate Limiter
- 5.5. Блок квантования по уровню Quantizer
- 5.6. Блок сухого и вязкого трения Coulomb and Viscous Friction
- 5.7. Блок люфта Backlash
- 5.8. Блок переключателя Switch
- 5.9. Блок многовходового переключателя Multiport Switch
- 5.10. Блок ручного переключателя Manual Switch

6. Math — блоки математических операций

- 6.1. Блок вычисления модуля Abs
- 6.2. Блок вычисления суммы Sum
- 6.3. Блок умножения Product
- 6.4. Блок определения знака сигнала Sign
- 6.5. Усилители Gain и Matrix Gain
- 6.6. Ползунковый регулятор Slider Gain
- 6.7. Блок скалярного умножения Dot Product
- 6.8. Блок вычисления математических функций Math Function
- 6.9. Блок вычисления тригонометрических функций Trigonometric Function
- 6.10. Блок вычисления действительной и (или) мнимой части комплексного числа Complex to Real-Imag
- 6.11. Блок вычисления модуля и (или) аргумента комплексного числа Complex to Magnitude-Angle
- 6.12. Блок вычисления комплексного числа по его действительной и мнимой части Real-Imag to Complex
- 6.13. Блок вычисления комплексного числа по его модулю и аргументу Magnitude-Angle to Complex
- 6.14. Блок определения минимального или максимального значения MinMax
- 6.15. Блок округления числового значения Rounding Function
- 6.16. Блок вычисления операции отношения Relational Operator
- 6.17. Блок логических операций Logical Operation
- 6.18. Блок побитовых логических операций Bitwise Logical Operator
- 6.19. Блок комбинаторной логики Combinatorial Logic
- 6.20. Блок алгебраического контура Algebraic Constraint

7. Signal&Systems — блоки преобразования сигналов и вспомогательные блоки

- 7.1. Мультиплексор (смеситель) Mux
- 7.2. Демультиплексор (разделитель) Demux
- 7.3. Блок шинного формирователя Bus Creator
- 7.4. Блок шинного селектора Bus Selector
- 7.5. Блок селектора Selector
- 7.6. Блок присвоения новых значений элементам массива Assignment
- 7.7. Блок объединения сигналов Merge
- 7.8. Блок объединения сигналов в матрицу Matrix Concatenation
- 7.9. Блок передачи сигнала Goto
- 7.10. Блок приема сигнала From
- 7.11. Блок признака видимости сигнала Goto Tag Visibility
- 7.12. Блок создания общей области памяти Data Store Memory
- 7.13. Блок записи данных в общую область памяти Data Store
- 7.14. Блок считывания данных из общей области памяти Data Store
- 7.15. Блок преобразования типа сигнала Data Type Conversion
- 7.16. Блок преобразования размерности сигнала Reshape
- 7.17. Блок определения размерности сигнала Width
- 7.18. Блок определения момента пересечения порогового значения Hit Crossing
- 7.19. Блок установки начального значения сигнала IC
- 7.20. Блок проверки сигнала Signal Specification
- 7.21. Датчик свойств сигнала Probe
- 7.22. Блок, задающий количество итераций Function-Call Generator
- 7.23. Информационный блок Model Info

8. Function & Tables — блоки функций и таблиц

- 8.1. Блок задания функции Fcn
- 8.2. Блок задания функции MATLAB Fcn
- 8.3. Блок задания степенного многочлена Polynomial
- 8.4. Блок одномерной таблицы Look-Up Table
- 8.5. Блок двумерной таблицы Look-Up Table(2D)
- 8.6. Блок многомерной таблицы Look-Up Table (n-D)
- 8.7. Блок таблицы с прямым доступом Direct Look-Up Table (n-D)
- 8.8. Блок работы с индексами PreLook-Up Index Search
- 8.9. Блок интерполяции табличной функции Interpolation (n-D) using PreLook-Up

9. Subsystem — подсистемы

- 9.1. Виртуальная и монолитная подсистемы Subsystem и Atomic Subsystem
- 9.2. Управляемая уровнем сигнала подсистема Enabled Subsystem
- 9.3. Управляемая фронтом сигнала подсистема Triggered Subsystem
- 9.4. Управляемая уровнем и фронтом сигнала подсистема Enabled and Triggered Subsystem
- 9.5. Управляемая S-функцией подсистема Function-call subsystem
- 9.6. Блок условного оператора If
- 9.7. Блок переключателя Switch Case
- 9.8. Управляемая по условию подсистема Action Subsystem
- 9.9. Управляемая подсистема For Iterator Subsystem