Государственное бюджетное образовательное учреждение высшего образования Московской области «Университет «Дубна» Филиал «Протвино» Кафедра «Автоматизация технологических процессов и производств»

В.А. Коковин, А.П. Леонов

ЛАБОРАТОРНЫЕ РАБОТЫ ПО ДИСЦИПЛИНЕ «ПРОЕКТИРОВАНИЕ СИСТЕМ УПРАВЛЕНИЯ»

ЭЛЕКТРОННОЕ МЕТОДИЧЕСКОЕ ПОСОБИЕ

Рекомендовано кафедрой автоматизации технологических процессов и производств филиала «Протвино» государственного университета «Дубна» в качестве методического пособия для студентов, обучающихся по направлению «Автоматизация технологических процессов и производств»

Рецензент:

кандидат технических наук, ведущий научный сотрудник ФГБУ «ИНСТИТУТ ФИЗИКИ ВЫСОКИХ ЭНЕРГИЙ имени А.А. Логунова Национального исследовательского центра «Курчатовский институт» Е.А. Устинов

Коковин, В.А.

К59 Лабораторные работы по дисциплине «Проектирование систем управления»: электронное методическое пособие / В.А. Коковин, А.П. Леонов — Протвино, 2016. — 64 с.: ил.

Методическое пособие содержит теоретические сведения и описание четырех лабораторных работ по дисциплине «Проектирование систем управления». Цель лабораторных работ – получение студентами практических навыков проектирования замкнутых систем управления с использованием пакетов моделирования, включая разработку моделей систем управления, исследование качества переходных процессов и их оптимизацию, настройку параметров ПИД-регуляторов, а также – освоение экспериментальных методов определения качественных показателей систем автоматического управления.

Методическое пособие предназначено для студентов, обучающихся по направлению «Автоматизация технологических процессов и производств».

ББК 32.965

© Государственное бюджетное образовательное учреждение высшего образования Московской области «Университет «Дубна», филиал «Протвино», 2016 © Коковин В.А., Леонов А.П.

Введение

Проектирование систем управления играет важную роль в современных технологических системах. Выгоды от совершенствования систем управления в промышленности могут быть огромны.

Проектирование системы управления технологическим оборудованием — сложный процесс, который требует принятия решений и использования итераций. Успех при проектировании систем управления зависит от двух ключевых моментов [1]:

а) всестороннего понимания самого автоматизируемого процесса,

б) знания фундаментальных концепций замкнутых систем.

Чтобы успешно спроектировать систему управления, прежде всего следует понять, как развивается процесс. В результате исследования процесса создается его математическая модель, позволяющая описать систему управления в динамическом и установившемся режимах. При построении модели обычно исходят из физических соображений в сочетании с экспериментальными данными. Модель может использоваться для синтеза регулятора. Одно из преимуществ математической модели заключается в том, что она может быть использована для моделирования ситуаций, которые в реальности могли бы привести к необратимым последствиям.

Лабораторная работа № 1

Разработка моделей систем управления на основе пакета визуального программирования *Simulink*

Цель работы — изучение основных библиотечных блоков пакета *Simulink*; освоение основных приемов создания и редактирования моделей; получение навыков задания параметров расчета и диагностирования моделей.

1 Сведения из теории [2,3]

1.1 Общие сведения

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

Simulink является достаточно самостоятельным инструментом *MATLAB*. 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*. Имеются 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, *NCD* -*Nonlinear Control Design Blockset* — моделирование электротехнических устройств, *Digital Signal Processing Blockset* — набор блоков для разработки цифровых устройств и т. д.).

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

При моделировании пользователь может выбирать метод решения дифференциальных уравнений, а также — способ изменения модельного времени (с фиксированным или переменным шагом).

В ходе моделирования есть 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.

Преимущество *Simulink* и в том, что он позволяет пополнять библиотеки блоков с помощью подпрограмм, написанных как на языке *MATLAB*, так и на языках *C++*, *Fortran* и *Ada*.

1.1 Создание модели

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

1.2.1. Создать новый файл модели с помощью команды *File / New / Model* или использовать кнопку на панели инструментов (здесь и далее символом "/"обозначены пункты меню программы, которые необходимо последовательно выбрать для выполнения указанного действия). Вновь созданное окно модели показано на рис. 1.1.



Рис 1.1. Пустое окно модели

1.2.2. 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» его в созданное окно. Клавишу «мыши» нужно держать нажатой. На рис. 1.2 показано окно модели, содержащее блоки.

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

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



Рис. 1.2. Окно модели, содержащее блоки

1.2.3. Чтобы изменить параметры блока, установленные программой «по умолчанию», нужно дважды щелкнуть левой клавишей «мыши», указав курсором на его изображение. Откроется окно редактирования параметров 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 *OK*. На рис. 1.3 в качестве примера показаны: блок, моделирующий передаточную функцию, и окно редактирования его параметров.

	Transfer Fon
	Matrix expression for numerator, vector expression for denominator. Output width equals the number of rows in the numerator. Coefficients are for descending powers of s.
1	Parameters
s+1	Numerator:
Transfer Fcn	[1]
	Denominator:
	[1 1]
	Absolute tolerance:
	auto
	OK Cancel <u>H</u> elp <u>Apply</u>

Рис. 1.3. Блок, моделирующий передаточную функцию и окно редактирования параметров блока

1.2.4. После установки всех блоков из библиотек нужно на схеме 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 этого отпустить клавишу. В случае правильного соединения изображение стрелки на входе блока изменит цвет. Для создания точки разветвления в соединительной линии нужно подвести курсор к предполагаемому узлу и, нажав <u>правую</u> клавишу «мыши», протянуть линию. Для удаления линии требуется выбрать линию (как это выполняется и для блока), а затем нажать клавишу **Delete** на клавиатуре. Схема модели, в которой выполнены соединения между блоками, показана на рис. 1.4. (Подробнее смотри раздел 1.3.6.)



Рис. 1.4. Соединение элементов схемы

1.2.5. 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 *Fille/Save*. При повторных запусках программы *SIMULINK* загрузка схемы осуществляется с помощью меню *File/Open...* в окне обозревателя библиотеки или из основного окна *MATLAB*.

1.3 Основные приемы подготовки и редактирования модели

1.3.1. 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. По аналогии можно изменить и подписи к блокам моделей. На рис. 1.5 показаны текстовая надпись и изменение надписи к блоку передаточной функции. Следует иметь в виду, что рассматриваемая версия программы (*Simulink* 4) не адаптирована к кириллическим шрифтам, а их применение может иметь непредсказуемые последствия: например, отображение надписей в нечитаемом виде, обрезание надписей, сообщения об ошибках, а также — невозможность открыть модель после ее сохранения. Поэтому использование надписей на русском языке для текущей версии *Simulink* крайне нежелательно.



Рис. 1.5. Текстовая надпись и ее изменение в *Transfer Function*

1.3.2. Выделение объектов.

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

Выделение объектов проще всего осуществляется «мышью»: установить курсор «мыши» на нужном объекте и щелкнуть левой клавишей «мыши». Произойдет его выделение. Об этом будут свидетельствовать маркеры по углам объекта (см. рис. 1.5). Можно также выделить несколько объектов: 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. Так можно выделить все объекты, 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.

1.3.3. 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 инструментом **h**а панели инструментов. Для вырезания объекта в буфер его необходимо предварительно выделить, а затем выполнить команду *Edit/Cut* или воспользоваться инструментом **k** на панели инструментов. При выполнении данных операций следует иметь в виду, что объекты помещаются в собственный буфер *MATLAB* и становятся недоступными в других приложениях. Использование команды *Edit/Copy model to Clipboard* позволяет поместить *графическое изображение* модели в буфер *Windows* и 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 перемещать в необходимое место.

1.3.4. 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.

1.3.5. 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*.

1.3.6. 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. 1.6). Держа

нажатой левую кнопку «мыши», нужно переместить курсор к входному порту нужного блока. Курсор «мыши» примет вид креста из тонких сдвоенных линий (рис. 1.7). После создания линии необходимо отпустить левую клавишу «мыши». Свидетельством того, что соединение создано, будет жирная стрелка у входного порта блока. Выделение линии производится точно так же, как и выделение блока — одинарным щелчком левой клавиши «мыши». Черные маркеры, расположенные в узлах соединительной линии, будут говорить о том, что линия выделена.



Рис. 1.6. Начало создания соединения

🖬 untitled *					- 🗆 ×
<u>File</u> <u>E</u> dit	_ <u>⊻</u> iew <u>S</u> imu	lation F <u>o</u> rmat	<u>T</u> ools <u>H</u>	lelp	
	5 X B 🛍	🗅 🗠 🕌 🖬	<u>⊫</u> 🛞 ► –	Normal	•
	1 Const] ant5	Cope		
Ready	150%		ode4	15	

Рис. 1.7. Завершение создания соединения

Петля линии соединения выполняется так же, как и перемещение блока. Линия соединения выделяется, а затем нужная часть линии перемещается. Рис. 1.8 поясняет этот процесс.

Удаление соединений выполняется так же, как и любых других объектов (см. п. 1.3.5).



Рис. 1.8. Создание петли в соединительной линии

1.3.7. Изменение размеров блоков.

Для изменения размера блок выделяется, после чего курсор "мыши" устанавливается на один из маркеров по углам изображения. После превращения курсора в двустороннюю стрелку, необходимо нажать левую клавишу «мыши» и растянуть (или сжать) изображения блока. На рис. 1.9 показан этот процесс. Размеры надписей блока при этом не изменяются.



Рис. 1.9. Изменение размера блока

1.3.8. Перемещение блоков.

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

1.3.9. Использование команд *Undo* и *Redo*.

В процессе освоения программы пользователь может совер-шать действия, кажущиеся ему необратимыми (например, случай-ное удаление части модели, копирование и т. д.). В этом случае следует воспользоваться командой *Undo* (это 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* (инструмент

). 1.3.10. Форматирования объектов.

В меню *Format* (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 colo — 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 90⁰ по часовой

стрелке; *Show drop shadow* — показ тени от блока;

Show port labels — показ меток портов.

4. Прочие установки:

Library link displa — показ связей с библиотеками;

Sample time colors — выбор цвета блока индикации времени;

Wide nonscalar lines — увеличение/уменьшение ширины нескалярных 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.

1.4 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 рис. 1.10.

📣 Sim	ulation Para	meters: un	titled		_ 🗆 X
Solver	Workspace I/O	Diagnostics	Advanced		
Simul Start	ation time time: 0.0	Stop tii	me: 10.0		
Solve Type:	er options Variable-step	▼ ode45	5 (Dormand-Pri	nce)	
Max	step size: auto		Relative tolera	ance: 1e-3	
Min s	tep size: auto		Absolute toler	ance: auto	
Initial	step size: auto				
Outpu	ut options				
Refir	ne output	•	Refine fa	actor: 1	
		OK	Cancel	Help	Apply

Рис 1.10. 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 *МАТLAB*;

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 (рис. 1.10): *Simulation time* (интервал моделирования или, иными словами, время расчета), *Solver options* (параметры расчета), *Output options* (параметры вы-вода).

1.4.1. Установка параметров расчета модели.

1.4.1.1. *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.

1.4.1.2. *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 переменным (*Variable-step*) шагом. Как правило, *Variable-step* используется для моделирования непрерывных систем, а *Fixed-step* — дискретных.

Список методов расчета нового состояния системы содержит несколько вариантов: первый (*discrete*) используется для расчета дискретных систем, остальные — 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 *Туре* 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 (рис. 1.11).

Величина шага моделирования устанавливается системой по умолчанию 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) — используется, если в модели есть 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. *Auto* (Автоматический выбор режима) — позволяет *Simulink* автоматически устанавливать режим **MultiTasking** для тех моделей, в которых используются блоки с различными скоростями передачи сигналов, а режим *SingleTasking* — для моделей с блоками, оперирующими одинаковыми скоростями.

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

🤌 Sim	ulation Parar	neters: un	titled			_ 🗆 X
Solver	Workspace I/O	Diagnostics	Advar	nced		
Simul Start	ation time time: 0.0	Stop tin	ne: 10	.0	_	
Solve Type:	r options Fixed-step	discret	e (no c	ontinuou:	s states)	J
Fixed	step size: auto		Mode:	Auto Auto SingleT MultiTa:	∎ asking sking	
Outpu	ut options					19 19
Refir	ne output	7	Be	efine fact	or: 1	
		ОК	Car	ncel	Help	Apply

Рис. 1.11. Вкладка *Solver* при выборе фиксированного шага расчета

Max step size — максимальный шаг расчета. По умолчанию он устанавливается автоматически (*auto*) и его значение *Stop* в этом случае равно (*StopTime* — *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 пе-

ременного шага необходимо указать точность вычисле-

ний: относительную (*Relative tolerance*) и абсолютную (*Absolute tolerance*). По умолчанию они равны, соответственно, 10^{-3} и *auto*.

1.4.1.3. Параметры вывода (Output options).

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 — значит 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 и 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).

1.5 Установка параметров обмена с рабочей областью

Элементы, позволяющие управлять вводом и выводом промежуточных данных и результатов моделирования в рабочую область *MATLAB*, расположены на вкладке *Workspace I/O* (рис. 1.12).

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 бу-

дут 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*).

olver Workspace I/O D	iagnostics Advanced	
Load from workspace	Save to wo	rkspace tout xout yout
Save options 「「Limit data points to last: Decimation:	1000	3.G. J The in con

Рис. 1.12. 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) — Задают количество строк при передаче переменных в рабочую область. Если флажок Limit rows to last установлен, то в поле ввода можно указать коли-чество 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. Доступ-

ные форматы Array (массив), Structure (структура),

Structure With Tim (структура с дополнительным полем – «время»).

1.6 Установка параметров диагностирования модели

Вкладка *Diagnostics* (рис. 1.13) 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.

1.7 Выполнение расчета

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

1.8 Завершение работы

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

Solver Workspace 1/1	Diagnostics	Advanced	
Simulation options Consistency checking	r none 💌	Bounds checki	ng: none 💌
-1 sample time Algebraic loop Block priority Check for singu Data overflow int32 to float Invalid FcnCall Min step size v MultiTask rate S-function upor	in source violation lar matrix conversion connection iolation transition ades needed	Warning Warning None Warning Warning Error Warning Error None	Action C None C Warning C Error
	ОК	Cancel	Help Apply

Рис. 1.13. Вкладка *Diagnostics* окна установки параметров моделирования

2Основные библиотечные блоки Simulink 3

2.1 Sources — источники сигналов

2.1.1 Источник постоянного сигнала Constant

Назначение — задание постоянного по уровню сигнала. **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.

Рис. 1.14 иллюстрирует применение этого источника и измере-ние его выходного сигнала с помощью цифрового индикатора *Display*.

3.14	3.14	Block Parameters: Constant3
Constant	Display	Constant
sin(3.14) Constant1	0.001593	Output the constant specified by the 'Constant value' parameter. If 'Constant value' is a vector and 'Interpret vector parameters as 1-D' is on, treat the constant value as a 1-D array. Otherwise, output a matrix with the same dimensions as the constant value.
[-0.5 1 1.5]	-0.5 1 1.5	Parameters Constant value:
	Display2	Interpret vector parameters as 1-D
Constant3	1.5 -3.7 Display3	OK Cancel Help Apply
3.14+4i	3.14 + 4i	
Constant4	Display4	

Рис. 1.14. Источник постоянного воздействия Constant

2.1.2 Источник синусоидального сигнала Sine Wave

Назначение — формирование синусоидального сигнала с заданной частотой, амплитудой, фазой и смещением.

Для формирования выходного сигнала блоком могут использоваться два алгоритма. Вид алгоритма определяется параметром *Sine Type* (способ формирования сигнала):

Time-based — по текущему времени; *Sample-based* —

по величине шага модельного времени.

2.1.2.1. Формирование выходного сигнала по текущему значению времени для непрерывных систем.

Выходной сигнал источника в этом режиме соответствует выражению:

 $y = Amplitude \times +sin(frequency time + phase) bias$

Параметры: *Amplitude* — амплитуда; *Frequency* (*rads/sec*) — частота (paд/c); *Phase* (*rads*) — начальная фаза (paд); *Bias* — постоянная составляющая сигнала.

Sampl time — шаг модельного времени. Используется для согласования работы источника и других компонентов модели во времени. Параметр может принимать следующие значения:

Sample time = 0 (по умолчанию). Используется при моделировании непрерывных систем.

Sample time > 0 (положительное значение). Задается при моделировании дискретных систем. В этом случае шаг модельного времени можно интерпретировать как шаг квантования по времени выходного сигнала.

Sample time = -1. Шаг модельного времени устанавливается таким же, как и в предыдущем блоке, откуда приходит сигнал в данный блок. Этот параметр может задаваться для большинства блоков библиотеки Simulink. В дальнейшем он рассматриваться не будет.

При расчетах для очень больших значений времени точность расчета выходных значений сигнала падает вследствие значительной ошибки округления.

2.1.2.2. Формирование выходного сигнала по текущему времени для дискретных систем

Алгоритм определения значения выходного сигнала источника для каждого последующего шага расчета определяется выражени-

ем (в матричной форме):

 $\begin{vmatrix} \sin(t^{-1} \Lambda t) \\ \cos(t + \Delta t) \end{vmatrix} = \begin{vmatrix} \cos^{\Lambda} t & \sin \Delta t & \sin t \\ -\sin \Delta t & \cos \Delta t & \cos t \end{vmatrix}$

Где $y = t \Delta$ постоянная величина, равная значению Sample time.

В данном режиме ошибка округления для больших значений времени также уменьшает точность расчета.

2.1.2.3. Формирование выходного сигнала по величине модельного времени и количеству расчетных шагов на один период.

Выходной сигнал источника в этом режиме соответствует выражению:

 $y = Amplitude \ sin \qquad \frac{\# \ Number \ of \ offset \ samples}{Samples \ per \ period} + bias,$

где *k* — номер текущего шага расчета; *Amplitude* — амплитуда; Bias — постоянная составляющая сигнала; Samples_per_period количество расчетных шагов на один период синусоидального сигнала:

 $Samples_per_period = \frac{2}{frequency \times Sample_Time}$;

Sample_time — шаг модельного времени; *Num-ber_of_offset_samples* — начальная фаза сигнала. Задается количеством шагов модельного времени:

Number
$$_of_offset _samples = \frac{Phase Samples_per_period}{2\pi}$$
.

В данном режиме ошибка округления не накапливается, поскольку *Simulink* начинает отсчет номера текущего шага с нуля для каждого периода.

На рис. 1.15 показано применение блока с разными значениями шага модельного времени (*Sample time* = 0 для блока *Sine Wave* 1 и *Sample_time* = 0.1 для блока *Sine Wave* 2). Для отображения графиков выходных сигналов в модели использован виртуальный осциллограф *Scope*.



Рис. 1.15. Блок *Sine Wave*

2.1.3 Генератор ступенчатого сигнала *Step*

Назначение — формирование ступенчатого сигнала. Параметры:

Step time — время наступления перепада сигнала, с;

Initial value — начальное значение сигнала, В;

Final value — конечное значение сигнала, В.

Перепад напряжений льшую сторону (ко-нечное значение больше чем начальное), так и в меньшую (конеч-ное значение меньше чем начальное). Значения начального и ко-нечного уровней могут быть не только положительными, но и отрицательными (например, изменение сигнала с уровня 5В до уровня 3В).

На рис. 1.16. показано использование генератора ступенчатого сигнала.



Рис. 1.16. Блок *Step*

2.1.4 Источник импульсного сигнала Pulse Generator

Назначение — формирование прямоугольных импульсов. Параметры:

Pulse Type — способ формирования сигнала. Может принимать два значения:

Time-based — по текущему времени;

Sample-based — по величине модельного времени и количеству расчетных шагов;

Amplitude — амплитуда.

Period — период. Задается в секундах для *Time-based Pulse Туре* или в шагах модельного времени для *Sample-based Pulse Туре*;

Pulse width — ширина импульсов. Задается в в % по отношению к периоду для *Time-based Pulse Type* или в шагах модельного времени для *Sample-based Pulse Type*;

Phase delay — фазовая задержка. Задается в секундах для Timebased Pulse Type или в шагах модельного времени для Samplebased Pulse Type;

Sample time — шаг модельного времени. Задается для Samplebased Pulse Type.

Пример использования *Pulse Generator* показан на рис. 1.17.



Рис. 1.17. Источник прямоугольных импульсов

2.2 Sinks — приемники сигналов

2.2.1. Осциллограф Scope

Назначение — построение графиков исследуемых сигналов в функции времени. Позволяет наблюдать за изменениями сигналов в процессе моделирования.

Изображение блока и окно для просмотра графиков показаны на рис. 1.18.



Рис. 1.18. 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). Если на вход блока поступает векторный сигнал, кривая для каждого элемента вектора строится отдельным цветом.

Настройка окна осциллографа выполняется с помощью панелей инструментов (рис. 1.19).

9		P	ø	Ø	Ä	1	13		9	-15
1	2	3	4	5	6	7	8	9	10	11

Рис. 1.19 Панель инструментов блока Scope

Панель инструментов содержит 11 кнопок:

1-print (печать содержимого окна осциллографа);

2-parameter (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 f*loating scope*).

Изменение масштабов отображаемых графиков можно выполнять несколькими способами:

1. Нажать соответствующую кнопку (\wp , \wp или \wp) и щелкнуть один раз левой клавишей «мыши» в нужном месте графика. Про-изойдет 2,5 кратное увеличение масштаба.

2. Нажать соответствующую кнопку (\mathcal{P} , \mathcal{P} или \mathcal{P}) и, нажав левую клавишу «мыши», с помощью динамической рамки или отрезка указать область графика для увеличенного изображения. Рис. 1.20 поясняет этот процесс.



Рис. 1.20. Увеличение масштаба графика

3. 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 рис. 1.21.

Y-min: -1.1 Y-max: 1.1 Title ('% <signallabel>' replaced by signal name): %<signallabel></signallabel></signallabel>) 'Scope' p	roperties:	axis 1	
Title ('% <signallabel>' replaced by signal name): %<signallabel></signallabel></signallabel>	Y-min: -1.1		Y-max: 1.	1
% <signallabel></signallabel>				
	Title ('% <signa< td=""><td>ILabel>' replac</td><td>ed by signa</td><td>l name):</td></signa<>	ILabel>' replac	ed by signa	l name):

Рис. 1.21. Окно свойств графика

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 рис. 1.22.

'Scope' parameters	
General Data history Ti	p: try right clicking on axes
Axes Number of axes: 1 Time range: 2 Tick labels: bottom axis only	☐ floating scope
Sampling Decimation 💌 1	
OK Cancel	Help Apply

Рис. 1.22. 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. Задает режим вывода расчетных точек на экран. При выборе *Decimation* кратность вывода устанавливается числом, задающим шаг выводимых расчетных точек. На рис. 1.23 и 1.24. показаны графики синусоидальных сигналов, рассчитанных с фиксированным шагом, равным 0.1 с.



Рис. 1.23. Отображение синусоидального сигнала (*Decimation* = 1)

На рис. 1.23, в окне блока *Scope*, выводится каждая расчетная точка (параметр *Decimation* равен 1). На рис. 1.24 показан вывод каждого второго значения (параметр *Decimation* равен 2). Маркерами на графиках отмечены расчетные точки.



Рис. 1.24. Отображение синусоидального сигнала (*Decimation* = 2)

Если режим вывода расчетных точек задается как *Sample time*, его числовое значение определяет интервал квантования при отображении сигнала. На рис. 1.25 показан график синусоидального сигнала, для случая, когда значение параметра *Sample time* рав-но 0.1.

5. *Floating scope* — перевод осциллографа в «свободный» ре-жим (при установленном флажке).



Рис. 1.25. Отображение синусоидального сигнала (**Sample time** = 0.1)

На вкладке *Data history* (рис. 1.26) задаются следующие параметры:

🐠 'Floating S	Goope' p	oaramete	rs	
General Data	history	Tip: t	ry right clicki	ng on axes
🔽 Limit data p	points to la	ast: 5000		
Save data	to worksp	ace		
Variable name:	Scope)ata		
Format:	Structur	e with time		•
<u></u>		- 1	Г., ,	
	UK	Cancel	Help	Apply

Рис. 1.26. Вкладка Data history

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 дополни-тельным полем «время».

2.2.2 Осциллограф *Floating Scope*

Осциллограф *Floating Scope*, по сути, есть обычный осциллограф *Scope*, переведенный в «<u>свободный</u>» режим. Пример модели с осциллографом *Floating Scope* показан на рис. 1.27.



Рис. 1.27. Пример модели с осциллографом *Floating Scope*

В этом режиме блок осциллографа не имеет входов, а выбор отображаемого сигнала осуществляется с помощью инструмента **(Signal selection)** панели инструментов. Для выбора сигналов выполняются следующие действия:

1. Выделить систему координат, в которой будет отображаться график. Это достигается с помощью одиночного щелчка левой клавишей «мыши» внутри нужной системы. Выбранная система координат будет подсвечена (по периметру) синим цветом.

2. С помощью инструмента **т** открыть окно диалога *Signal Selector* (рис. 1.28).



Рис. 1.28. Окно диалога Signal Selector

3. Отметить флажком имена блоков, сигналы с выхода, которые требуется исследовать.

После выполнения расчета в окне блока *Floating Scope* будут отображены выбранные сигналы.

2.2.3 Графопостроитель XУ Graph

<u>Назначение</u> — построение графика одного сигнала в функции другого (график вида y = f(x).

Параметры:

х-тіп — минимальное значение сигнала по оси **X**;

х-тах — максимальное значение сигнала по оси **Х**;

у-тіп — минимальное значение сигнала по оси **Y**;

у-тах — 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 рис. 1.29, в качестве примера использования графопостроителя, показано построение фазовой траектории колебательного звена.



Рис. 1.29. Пример использования графопостроителя *XУ Graph*

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 рис. 1.30.



Рис. 1.30. Пример использования блока *XУ Graph* для отображения временных зависимостей

2.3 Continuous — аналоговые блоки

2.3.1 Блок вычисления производной Derivative

Назначение — выполнение численного дифференцирования входного сигнала.

Задаваемые параметры отсутствуют.

Для вычисления производной используется приближенная формула Эйлера $du_{dt} = \frac{\Delta U}{\Delta t}$, где ΔU — величина изменения входного сигнала за время Δt , Δt — текущее значение шага модельного времени.

Значение входного сигнала блока до начала расчета считается равным нулю. Начальное значение выходного сигнала также полагается равным нулю. Точность вычисления производной существенно зависит от величины установленного шага расчета. Выбор меньшего шага расчета улучшает точность вычисления производной.

На рис. 1.31 показан пример использования дифференцирующего блока для вычисления производной прямоугольного сигнала. В рассматриваемом примере, для повышения наглядности, шаг расчета выбран достаточно большим.



Рис. 1.31. Использование блока *Derivative* для дифференцирования сигнала

Данный блок используется для дифференцирования аналоговых сигналов. При дифференцировании дискретного сигнала с помощью блока *Derivative* его выходной сигнал будет представлять собой последовательность импульсов, соответствующих моментам времени скачкообразного изменения дискретного сигнала.

2.3.2 Интегрирующий блок Integrator

Назначение — выполнение интегрирования входного сигнала. Параметры:

external reset — внешний сброс. Тип внешнего управляющего сигнала, обеспечивающего сброс интегратора к начальному состоянию. Выбирается из списка:

none — нет (сброс не выполняется);

rising — нарастающий сигнал (передний фронт сигнала);

falling — спадающий сигнал (задний фронт сигнала);

either — нарастающий либо спадающий сигнал;

level — ненулевой сигнал (сброс выполняется, если сигнал на управляющем входе становится не равным нулю).

Если выбран какой-либо (но не *none*) тип управляющего сигнала, то на изображении блока появляется дополнительный управляющий вход. Рядом с дополнительным входом будет показано условное обозначение управляющего сигнала.

Initial condition source — источник начального значения выходного сигнала. Выбирается из списка:

initernal — внутренний;

external — внешний.

В этом случае на изображении блока появляется дополнительный вход, обозначенный *x*₀, на который необходимо подать сигнал, задающий начальное значение выходного сигнала интеграто-

pa.

Initial condition — начальное условие. Установка начального значения выходного сигнала интегратора. Параметр доступен, если выбран внутренний источник начального значения выходного сигнала.

Limit output (флажок) — использование ограничения выходного сигнала.

Upper saturation limit — верхний уровень ограничения выходного сигнала. Может быть задан как числом, так и символьной последовательностью *inf*.

Lower saturation limit — Нижний уровень ограничения выходного сигнала. Может быть задан как числом, так и символьной последовательностью *inf*.

Show saturation port — управляет отображением порта, выводящего сигнал, свидетельствующий о выходе интегратора на ограничение.

Выходной сигнал данного порта может принимать следующие значения:

ноль, если интегратор не находится на ограничении;

+ 1, если выходной сигнал интегратора достиг верхнего ограничивающего предела;

- 1, если выходной сигнал интегратора достиг нижнего ограничивающего предела.

Show state port (флажок) — отобразить/скрыть порт состояния блока. Данный порт используется, если выходной сигнал интегратора требуется подать в качестве сигнала обратной связи этого же интегратора. Например, при установке начальных условий через внешний порт или при сбросе интегратора через порт сброса. Выходной сигнал с этого порта может использоваться также для организации взаимодействия с управляемой подсистемой.

Absolute tolerance — абсолютная погрешность.

На рис. 1.32 показан пример работы интегратора при подаче на его вход ступенчатого сигнала. Начальное условие принято равным нулю.



Рис. 1.32. Интегрирование ступенчатого сигнала

Пример на рис. 1.33 отличается от предыдущего подачей начального значения через внешний порт. Начальное значение выходного сигнала в данном примере задано равным 10.



Рис. 1.33. Интегрирование ступенчатого сигнала с установкой начального значения выходного сигнал

Пример на рис. 1.34 демонстрирует использование входного порта для сброса выходного сигнала и порта состояния интегратора с целью организации обратной связи. Схема работает следующим образом: входной постоянный сигнал преобразуется интегратором в линейно-изменяющийся; по достижении выходным сигналом значения, равного 1, блок *Relational Operator* вырабатывает логический сигнал, по переднему фронту которого происходит сброс выходного сигнала интегратора до начального значения, равного нулю. В результате на выходе интегратора формируется пилообразный сигнал, изменяющийся от 0 до + 1.

Следующая схема (рис. 1.35) использует установку начального значения интегратора с помощью его выходного сигнала. В первый момент времени начальное значение выходного сигнала интегратора с помощью блока *IC* (*Initial Condition*) устанавливается равным нулю. По достижении выходным сигналом значения равного 1 блок *Relational Operator* подает сигнал сброса выходного сигнала интегратора на начальный уровень: при этом сигналом, задающим начальный уровень, оказывается инвертированный выходной сигнал интегратора (то есть — 1). Далее цикл работы схемы повторяется. В отличие от предыдущей схемы выходным сигналом генератора является двуполярный.



Рис. 1.34. Генератор пилообразного сигнала на основе интегратора



Рис. 1.35. Генератор двуполярного пилообразного сигнала на основе интегратора

2.3.3 Блок передаточной функции Transfer Fcn

<u>Назначение:</u> Блок передаточной характеристики *Transfer Fcn* задает передаточную функцию в виде отношения полиномов:

$$H(s) = \frac{Y(s)}{U(s)} = \frac{num(s)}{den(s)} = \frac{num(1)s^{nn-1} + num(2)s^{nn-2} + \dots + num(nn)}{den(1)s^{nd-1} + den(2)s^{nd-2} + \dots + den(nd)},$$

где *nn* и *nd* — порядок числителя и знаменателя передаточ-ной функции, *num* — вектор или матрица коэффициентов числи-теля, *den* — вектор коэффициентов знаменателя.

Параметры:

1. *Numerator* — вектор или матрица коэффициентов полинома числителя;

2. *Denominator* — вектор коэффициентов полинома знаменателя;

3. Absolute tolerance — абсолютная погрешность;

Порядок числителя не должен превышать порядок знаменателя.

Входной сигнал блока должен быть скалярным. Если коэффициенты числителя заданы вектором, то выходной сигнал блока бу-

дет также скалярным (как и входной сигнал). На рис. 1.36 показан пример моделирования колебательного звена с помощью блока *Transfer Fcn*.



Рис. 1.36. Пример моделирования колебательного звена

Если коэффициенты числителя заданы матрицей, то блок *Transfer Fcn* моделирует векторную передаточную функцию, которую можно интерпретировать как несколько передаточных функций, имеющих одинаковые полиномы знаменателя, но разные полиномы числителя. При этом выходной сигнал блока является векторным и количество строк матрицы числителя задает размерность выходного сигнала.

На рис. 1.37 показан пример блока *Transfer Fcn*, задающего векторную передаточную функцию. Там же показана модель, полностью аналогичная рассматриваемой по своим свойствам, но состоящая из отдельных блоков *Transfer Fcn*.

Начальные условия при использовании блока *Transfer Fcn* полагаются нулевыми. Если же требуется, чтобы начальные условия не были нулевыми, то необходимо с помощью функции *tf2ss* (инструмент *Control System Toolbox*) перейти от передаточной функции к модели в пространстве состояний и моделировать динамический объект с помощью блока.



Рис. 1.37. Пример моделирования векторной передаточной функции и ее аналог

3. Порядок выполнения лабораторной работы

1. Разработать структурную схему модели из следующих библиотечных блоков пакета *Simulink*: - генератор ступенчатого сигнала *Step*, библиотека *Sources*;

- сумматор, библиотека Math Operations;

- блок передаточной функции *Transfer Fcn*, библиотека Continuous;

- осциллограф (двухлучевой) Scope, библиотека Sinks.

2. Соединить один вход осциллографа с выходом модели, второй — с выходом генератора ступенчатого сигнала Step.

3. Задать численные значения параметров передаточной функции Transfer Fcn структурной схемы в соответствии с приведенными в таблице 1 вариантами передаточной функции.

4. Задать численные значения параметров генератора ступенчатого сигнала *Step*.

5. Установить параметры расчета модели (пункт 2.3.3).

6. Запустить модель на симуляцию.

7. Зафиксировать переходный процесс, отображаемый на осциллографе.

8. Проделать п.п. 3.3—3.7 с различными значениями передаточной функции *Transfer Fcn*.

Варианты пере	едаточных функций	Таблица 1
№ бригады	Вариант №1	Вариант №2
1	$\frac{1}{22s^2 + 4s + 1}$	$\frac{1}{4s^{5}+12s^{2}+0.7s+1}$
2	$\frac{1}{\frac{5}{8s} + 17s + 6s + 1}$	$\frac{1}{3s^2 + 0.3s + 1}$
3	$\frac{1}{0.8s^3 + 0.02s^2 + s + 1}$	$\frac{1}{0.6s^{3}+s^{2}+0.3s^{+}1}$
4	$\frac{1}{23s^3 + 1.1s^2 + s + 1}$	$\frac{1}{s^3 + 0.7s^2 + 0.06s + 1}$
5	$\frac{1}{12s^3 + 5s^2 + 2s + 1}$	$\frac{1}{0.14s^3 + s^2 + 0.3s + 1}$
6	$\frac{1}{0.6s^3 + s^2 + 0.3s + 1}$	$\frac{1}{7s^3 + 9s^2 + s + 1}$

1 V

4 Требования к выполнению отчета

Отчет по лабораторной работе выполняется в отдельной тетради и должен содержать:

1. Название лабораторной работы, ее цель, задачи.

2. Схемы, выполняемые в *Simulink*.

3. Полученные в результате моделирования осциллограммы.

5 Вопросы для самопроверки

1. Как настроить параметры модели в пакете *Simulink*.

2. Основные блоки библиотек Sources, Continuous и Sinks.

3. Настройка параметров осциллографа Scope и графопостроителя *XY Graph*.

4. Назначение и основные параметры передаточной функции *Transfer Fcn.*

Лабораторная работа № 2

Использование специализированного пакета *NCD Simulink* для определения параметров системы управления, обеспечивающих заданное качество регулирования

Цель работы — получение практических навыков определения оптимальных параметров системы управления, обеспечивающих заданное качество регулирования.

1Некоторые сведения из теории [2, 3, 4]

1.1 Описание инструментального пакета (ИП) Nonlinear Control Design Blockset (NCD-Blockset)

Инструментальный пакет (ИП) Nonlinear Control Design Blockset (NCD-Blockset) предоставляет в распоряжение пользователя графический интерфейс для настройки параметров динамических объектов, обеспечивающих желаемое качество переходных процессов. Средством для достижения указанной цели принимается)оптимизационный подход, обеспечивающий минимизацию функции штрафа за нарушение динамических ограничений. При помощи данного инструмента можно настраивать параметры нелинейной *Simulink*-модели, в качестве которых может быть заявлено любое количество переменных, включая скаляры, векторы и матрицы. Особую значимость имеет то обстоятельство, что в процессе настройки могут учитываться неопределенности параметрического типа математической модели, что позволяет синтезировать робастные законы управления.

Задание динамических ограничений осуществляется в визуальном режиме. На базе этих ограничений *NCD-Blockset* автоматически генерирует задачу конечномерной оптимизации так, чтобы точка экстремума в пространстве настраиваемых параметров соответствовала выполнению всех требований, предъявляемых к качеству процесса. Эта задача решается с привлечением специализированной процедуры квадратичного программирования из пакета *Optimization Toolbox*. Ход оптимизации контролируется на экране с помощью отображения графика контролируемого процесса и текущих значений минимизируемой функции. По завершении процесса его результат фиксируется в рабочем пространстве.

Рассмотрим простой пример, иллюстрирующий начальные основы использования ИП *NCD Blockset*.

Пусть требуется построить управление с обратной связью (регулятор температуры) для объекта, имеющего передаточную ϕ ункцию $G(s) = \frac{1}{(s+1)^2}$, так, чтобы замкнутая система имела перерегулирование менее 5% и время переходного процесса (входа в 1% зону установившегося значения) — менее 2 секунд. В качестве устройства управления будем использовать ПИД-регулятор с передаточной функцией вида $R(s) = K_p + K_i \frac{1}{s} + K_d s$. Задача состоит в том, чтобы определить такие коэффициенты K_p , K_i , K_d , которые обеспечивают заданные требования к качеству переходного процесса.

Построим Simulink-модель объекта управления в соответствии со схемой, представленной на рис. 2.1, поместив в окно модели блоки Constant (из библиотеки Sources), Sum (из библиотеки Math Operations), Transfer Fcn (из библиотеки Continuous), PID Controller (из библиотеки Simulink Extras / Additional Linear).



Рис. 2.1. Simulink-модель объекта управления

Настроим модель объекта, изменив параметр *Denominator* блока *Transfer Fcn*. При этом зададим в качестве его значения вектор [1 2 1]: рис. 2.2.

Matrix expression for numerator, vector expressi Output width equals the number of rows in the r for descending powers of s. Parameters Numerator: [11]	on for denominator. umerator. Coefficients are
Parameters Numerator: [11]	
Numerator:	
[1]	
- June]
Denominator:	
[1 2 1]	
Absolute tolerance:	
auto	
	2.

Рис. 2.2. Блок параметров Transfer Fcn.

Инициализируем в командном окне *Matlab* переменные *Kp=1*, *Ki=1*, *Kd=0*; настроим параметры блока *PID Controller* (см. рис. 2.3), вводя в поле параметра *Proportional* переменную *Kp*, в поле *Integral* — *Ki*, а в *поле Derivative* — *Kd*.

PID Controller (mask) (link)	
Enter expressions for proportional, integral, and derivative te P+I/s+Ds	erms.
Parameters Proportional:	
Кр	
Integral:	
Кі	
Derivative:	
Kd	
OK Cancel <u>H</u> elp	Apply

Рис. 2.3. Блок параметров *PID* — контроллера.

Таким образом, мы сформировали *Simulink*-модель объекта управления и теперь можем приступить к заданию ограничений, налагаемых на выход системы (блока *Transfer Fcn*).

Дважды щелкнув по блоку *NCD Outport*, получим окно, представленное на рис. 2.4.



Рис. 2.4. Основное окно блока NCD Outport

Установим коридор, в пределах которого должен находиться входной сигнал блока *NCD Outport* в соответствии с требованиями задачи. Это можно сделать при помощи «мыши», передвигая красные линии, являющиеся границами коридора. Местоположение этих линий можно установить точно (в невизуальном режиме) при помощи диалоговой панели *Constraint Editor*, возникающей при щелчке правой кнопкой «мыши» по линии (рис. 2.5).

🥠 Constraint Editor			
For selected constraint in Figure: 9 C Limiting constraint	iystem: untitled, Outport: 1		
Overachieving constraint	Weight (0-1) 1		
Type: none	Clear Type		
Position editor (x1 y1 x2 y2):			
[0 1.05 2 1.05]			
Done	1		

Рис. 2.5. Диалоговая панель Constraint Editor

После выполнения установок границ коридора (в соответствии с заданными значениями качества переходного процесса) окно должно выглядеть так, как это показано на рис. 2.6.



Рис. 2.6. Основное окно блока *NCD Outport* с заданными параметрами

Далее выберем пункт *Parameters*... меню *Optimization*. При этом откроется окно, в котором необходимо перечислить имена настраиваемых переменных *Kp*, *Ki*, *Kd* в поле *Tunable Variables* (рис. 2.7). В этом окне также изменим значение поля *Discretization interval* на 0.1 и поставим «галочку» напротив поля *Stop optimization as soon as the constraints are achieved* (для прекращения процесса оптимизации после выполнения всех ограничений). После внесения этих изменений нажимаем кнопку *Done*.

Tunable Variables:	Kp, Ki, Kd	Done
Lower bounds (optional):		Revert
Upper bounds (optional):		Help
Discretization interval:	0.1	
Variable Tolerance:	0.001	
Constraint Tolerance:		
Display optimization inform	nation	
Stop optimization as soor	as the constraints are achieved	
Compute gradients with b	etter accuracy (slower)	

Рис. 2.7. Окно задания параметров для оптимизации

Теперь все готово для начала процесса оптимизации. Нажмем на кнопку *Start* и понаблюдаем за развитием процесса. Для каждого этапа оптимизации в окне отображаются графики сигнала, соответствующие начальным (белый цвет) и текущим (зеленый цвет) значениям настраиваемых параметров (рис. 2.8). В командном окне *MATLAB* отображается информация о ходе оптимизации.



Рис. 2.8. Окно блока *NCD Outport* с графиками переходного процесса

По окончании процесса оптимизации, оптимальные значения настраиваемых переменных, соответствующие кривой зеленого цвета, сохраняются в рабочем пространстве MATLAB. В данном случае Кр = 8.3616; Кd = 2.6012; Кi = 1.8190.

Как было отмечено выше, NCD позволяет учитывать в процессе настройки математической модели неопределенности параметрического типа. Предположим, что наш объект управления подвержен значительным изменениям — так, что передаточная функция

объекта имеет вид
$$W(s) = \frac{K}{a^2 + 2a + 1}$$

Требуется синтезировать робастное управление, т. е. управление, обеспечивающее выполнение требований, предъявляемых к замкнутой системе при любых значениях параметров К и а в указанных диапазонах.

Для решения поставленной задачи инициализируем в командном окне MATLAB переменные a = 1 и K = 1 и изменим параметры блока Transfer Fcn: параметр Numerator — [K], параметр Denominator — [a*a 2*a 1] (рис. 2.9).

Nock Parameters: Transfer Fcn	Þ
Transfer Fcn	
Matrix expression for numerator, vector expression for deno Output width equals the number of rows in the numerator, for descending powers of s.	minator. Coefficients are
Parameters	
Numerator:	
[K]	
Denominator:	
[a*a 2*a 1]	
Absolute tolerance:	
auto	

Рис. 2.9. Задание параметров передаточной функции *Transfer Fcn*

Выберем пункт Uncertainty меню Optimization окна блока NCD Outport. При этом появится окно Uncertain Variables, предназначенное для задания неопределенностей модели. Здесь необходимо указать имена переменных, значения которых варьируются, их верхние и нижние пределы изменения, а также указать какие пределы неопределенных переменных учитывать при параметрической оптимизации. Установим следующие значения (рис. 2.10):

поле *Uncertain variables*: a, K; поле *Lower bounds*: 0.5, 1; поле *Upper bounds*: 1, 2.

🜖 Uncertain Variables	_ 🗆 🗙
Uncertain variables:	
a, K	
Lower bounds:	
0.5, 1	
Upper bounds:	
1.2	
Number of Monte Carlo simulations:	
0	
Constrain nominal simulation	
Constrain lower bound simulation	
Constrain upper bound simulation	
Constrain Monte Carlo simulations	
Total simulations per step = 3	
Done	Help

Рис. 2.10. Окно *Uncertain Variables* для задания неопределенностей модели

Поставим «галочки» напротив полей Constrain lower bound simulation и Constrain upper bound simulation (будем учитывать при моделировании и верхние, и нижние допустимые пределы параметров a и K).

Теперь можно приступать к запуску модели, то есть начинать процесс оптимизации с помощью пакета *NCD*. При этом программа учитывает параметры оптимизации, заданные к началу процесса оптимизации (рис. 2.11).



Рис. 2.11. Графики переходных процессов системы регулирования

По окончании процесса оптимизации оптимальные значения настраиваемых переменных, соответствующие кривым зеленого цвета, сохраняются в рабочем пространстве *MATLAB*: Kp = 20.6148; Kd = 2.8250; Ki = 4.3390.

1.2 Схема применения блока NCD Blockset

Как видно из рассмотренных выше примеровах, при использовании *NCD Blockset* необходимо выполнить определенную последовательность действий:

1. Создать *Simulink*-модель.

2. Подсоединить *NCD* блок (или блоки) к сигналу (сигналам), на который будут накладываться ограничения. (Для открытия кна библиотеки *NCD Blockset* можно выполнить команду в окне *MATLAB - ncdblock*).

3. Задать в рабочем пространстве *MATLAB* начальные значения настраиваемых параметров.

4. Открыть *NCD* блок в *Simulink*-модели. При этом откроется окно, в котором в визуальном режиме формируются допустимые «коридоры» для каждого контролируемого сигнала.

5. Сформировать указанные коридоры путем растяжения, перемещения, разбиения или объединения граничных ломаных, используя команды меню *Edit*, кнопки «мыши» или «горячие клавиши». Допустимые коридоры можно сохранять и загружать из файла.

6. Открыть диалог *Optimization Parameters* при помощи команды *Parameters*... из меню *Optimization*. Указать интервал дискретизации (разработчики рекомендуют выбирать его в интервале между 1 и 2 процентами от общего времени интегрирования). Указать имена настраиваемых параметров, разделяя их пробелами или запятыми. Там же указать точность задания настраиваемых переменных и ограничений.

7. Если нужно — открыть диалог Uncertain Variables, выбрав команду Uncertainty меню Optimization для задания допустимых границ параметров неопределенности модели. Указанные параметры должны быть инициализированы в рабочем пространстве MATLAB.

8. Если нужно — сохранить все установки блока с использованием команды *Save*... меню *File*. Сохраненные установки могут использоваться в дальнейшем: их загрузка осуществляется при помощи команды *Load*... меню *File*.

9. Нажать кнопку *Start* или выбрать команду *Start* из меню *Optimization* для запуска процесса оптимизации.

10. Наблюдать за развитием процесса на экране.

11. По окончании процесса оптимизации настроенные значения параметров фиксируются в рабочем пространстве *MATLAB*.

2 Порядок выполнения лабораторной работы

1. Разработать структурную схему модели из следующих библиотечных блоков пакета *Simulink*:

- генератор ступенчатого сигнала Step, библиотека

Sources; - сумматор, библиотека Math Operations;

- блок передаточной функции *Transfer Fcn*, библиотека *Continuous*;

- осциллограф (двухлучевой) *Scope*, библиотека *Sinks*;

- блок оптимизации NCD Outport1.

2. Соединить один вход осциллографа с выходом модели, второй — с выходом генератора ступенчатого сигнала *Step*.

3. Соединить вход блока оптимизации *NCD Outport1* с выходом модели.

4. Задать численные значения параметров передаточной функции *Transfer Fcn* в соответствии с приведенным в таблице 1 вариантом N¹. При этом указанные параметры должны быть инициализированы в рабочем пространстве *MATLAB*.

5. Задать численные значения параметров генератора ступенчатого сигнала *Step*.

6. Установить параметры расчета модели.

7. Запустить модель на оптимизацию.

8. Зафиксировать график переходного процесса, отображаемый в окне блока *NCD Outport1* и на осциллографе.

9. Задать численные значения параметров передаточной функции *Transfer Fcn* в соответствии с приведенным в таблице 1 вариантом № 2. Проделать п.п. 6—8.

Значения перелаточных	функций	Transfer	Fcn	Таблица 1
эпи юпил передите шых	функции	I i unsjei		таолица т

	pedate men af main 1 a	изуст тем таблица т
	Вариант №1	Вариант №2
Бригада	K	<i>K</i>
№ 1	$(T1*T2) p^{2}+4 p+1$	$\overline{1.2*(T1*T2)p^2+0.7p+1}$
Бригада	K	K
Nº2	$7 * (T1*T2) p^{2}+6p +1$	$(3 p * (T1 * T2)^{2} + 0.3 p + 1)$
Бригада №3	$\frac{K}{0.02 * (T1*T2)p^{2} + p + 1}$	$\frac{K}{0.6 * (T1 * T 2) p^{2} + 0.3 p + 1}$
Бригада	K	K
<u>№</u> 4	$5^{*}(T1 * T2) p^{2} + p + 1$	$7*(T1*T2)p^{2}+0.06p+1$
Бригада	K	<i>K</i>
№ 5	$10 * (T1 * T2) p^{2} + 2 p + 1$	$9*(T1*T2)p^{2}+0.3p+1$
Бригада №6	$\frac{K}{9 (T1 \ T_{*} 2) p^{2} + 0.3 p + 1}$	$\frac{K}{9_{*}(T1 \ *T \ 2) \ p^{2} \ + p \ +1}$

4 Требования к выполнению отчета

Отчет по лабораторной работе выполняется в отдельной тетради и должен содержать:

1. Название лабораторной работы, ее цель, задачи.

2. Схемы, выполняемые в *Simulink*.

3. Полученные в результате моделирования осциллограммы.

5 Вопросы для самопроверки

1. Как настроить параметры оптимизации модели в пакете *NCD*, *Simulink*.

2. Основные блоки библиотек Sources, Continuous u Sinks.

3. Настройка неопределенных параметров оптимизации блока *NCD Outport1*.

4. Последовательность действий при оптимизации системы управления с помощью пакета *NCD*, *Simulink*.

Лабораторная работа № 3

Настройка параметров ПИД-регуляторов с помощью пакета Simulink

Цель работы — освоить методику настройки параметров ПИД-регуляторов с помощью пакета *Simulink*.

1 Сведения из теории [3, 4, 5]

Качество процессов управления в значительной мере определяется законом (алгоритмом) управления, реализуемым в САУ. В промышленных САУ регуляторы обычно проектируют как автономные устройства, реализующие определенный алгоритм (алгоритмы) управления. Структурная схема САУ для этого случая представлена на рис. 3.1, где X(p) — входное (задающее) воздействие, Y(p) — выходная (регулируемая) величина, $\Delta X(p) = X(p) Y(p)$ — ошибка регулирования, $W_{OO}(p)$ передаточная функция устройства управления УУ (автономного регулятора), U(p) — управляющее воздействие, $W_{OO}(p)$ — суммарная передаточная функция всех остальных звеньев САУ, кроме УУ. Поскольку алгоритм управления формируется в зависимости от W_{OY} (p) и заданных значений показателей качества (ошибка, время переходного процесса, перерегулирование и т.д.), то для настройки промышленных регуляторов необходимо иметь:

1. Набор настраиваемых параметров (коэффициентов) УУ;

2. Методику определения величин настраиваемых параметров УУ.



Рис. 3.1. Структурная схема САУ с автономным регулятором

Широкое применение в промышленности получили ПИД-регуляторы. Они реализуют пропорционально–интегрально–дифференциальный закон управления

$$u(t) \neq \mathbf{K}_p \quad (t) \Delta (t) \quad \mathbf{K}_i \quad \int \Delta x(t) + \mathbf{K}_d \quad \frac{d\Delta x}{dt} \tag{1},$$

где настраиваемыми параметрами являются коэффициенты пропорциональной части K_p , интегральной — K_i и дифференциальной — K_d .

В операторной форме передаточная функция ПИД–регулятора будет иметь вид

$$\frac{W_{VV(p)}}{\Delta x(p)} = \frac{u(p)}{\Delta x(p)} = K_p + \frac{K_i}{p} + K_d p(1 + T_d p)$$

Наличие частотно-зависимых — интегральной (И) и дифференциальной (Д) составляющих — обеспечивает устойчивость и минимизирует статическую и динамическую погрешности САУ. При этом И-составляющая будет вносить основное воздействие в закон управления на низких частотах, а Д-составляющая — на высоких. Иногда используют модифицированный ПИД-алгоритм, когда в закон (1) вместо $K_d \overset{d}{\xrightarrow{x}} \frac{x}{d}$ вводят сигнал производной от выходного $\frac{dt}{dt}$

(регулируемого) параметра *К* d <u>dt</u>. В ряде случаев, при скачкооб-

разных изменениях задающего воздействия *x*_{зад}, это улучшает показатели переходного процесса.

Настройка ПИД-регулятора заключается в определении значений коэффициентов K_p , K_i , K_d . Для каждой конкретной системы эти коэффициенты имеют различные значения, определяемые:

— параметрами остальных звеньев САУ, кроме ПИД– регулятора, т.е. $W_{OY}(p)$;

— заданными величинами показателей качества (статическая ошибка $_{ycm}$, перерегулирование, время переходного процесса t_{nn} и др.). ε

Для настройки ПИД-регулятора удобно использовать пакет *Simulink* <u>в режиме оптимизации.</u> В *Simulink Matlab* 6.1 из библиотечных элементов строится структурная схема (рис. 3.2). Для ее построения используем библиотечные элементы из *Simulink Library Browser*:

- элемент *Step* (1(t) — генератор единичной функции) из *Sim-ulink/Sourses*;

- Sum из Simulink/Math;

- PID Controller us Simulink Extras/Additional Linear;

ОУ состоит из 3-х модулей — Saturation из Simulink/Nonlinear (нелинейность типа «Ограничение», выбираем уровни ограничения -2 В и + 2 В), Rate Limiter из Simulink/Nonlinear (нелинейность динамического ограничения коэффициента усиления, выбираем ограничение в диапазоне [-0.8; +0.8]);

модуль *Transfer Fcn* с передаточной функцией W(s) из *Sim*-

ulink/Continuous, задаем = $\frac{W(s)}{50s^3 + 43s^2}$ 3s 1

- осциллограф Scope из Simulink/Sinks;

- блок задания ограничений NCD Outport из NCD Blockset.



Рис. 3.2. Пример структурной схемы САУ с ПИД-регулятором

Чтобы посмотреть и изменить передаточную функцию модуля надо дважды кликнуть на него "мышкой" и в появившемся меню произвести соответствующие изменения.

Постановка задачи настройки такова: при заданной структуре САУ необходимо найти значения коэффициентов K_p , K_i . K_d , при которых переходный процесс будет иметь:

- наибольшее перерегулирование $\sigma \sigma$ — не более 20%;

- длительность переходного процесса t_{nn} t_{nn} — не более 30 с.;

- время нарастания — не более 10 с.;

- установившуюся ошибку ε_{vcm} ε_{yct} — не более 5%.

Заданные значения устанавливают в блоке *NCD Outport*, формируя коридор для требуемого переходного процесса.

<u>Начальные значения коэффициентов К_p, К_i, К_d выбираются в соответствии с методикой настройки ПИД–регуляторов Зиглера–</u><u>Николса (*Ziegler–Nickols method*)</u>, в соответствии с которой:

1. Коэффициенты K_i . K_d устанавливаются равными нулю, а коэффициент K_p изменяется до тех пор, пока в системе не установятся автоколебания (колебания на экране *Scope* будут иметь постоянную амплитуду). Перед нажатием *Start* необходимо в меню Simulation установить время моделирования, для данного приме-ра — 100 с., а для *Scope*, дважды кликнув на него — масштабы осей.

2. Полученное предельное значение K_p обозначается как $K_{\ddot{O}}$ ($K_p = K_{\mathcal{U}}$), а период автоколебаний — как $P_{\mathcal{U}}$. Для данного примера $K_p = K_{\mathcal{U}} = 1.05$; $P_{\ddot{O}} = 26.8$ с.

3. Задаются следующие начальные значения коэффициентов ПИД-регулятора:

$$K_{p} = \frac{3K_{II}}{5} = 0.63; K_{i} = \frac{6K_{II}}{5P_{II}} = 0.05; K_{d} = \frac{3K_{II}P_{II}}{40} = 2.1$$

Начальные значения задаются в командной строке *Matlab*, значения отделяются запятой. Например, $K_i = 0.05$, $K_d = 2.1$.

4. Дважды кликнув на блок PID Controller. устанавливаем буквенные значения коэффициентов: *Proportional: K*_p*Integral: K*_i

Derivative: K_d

5. Дважды кликнув на *NCD Outport*, выбираем в меню *Optimization Parameters*. В открывшемся окне в строке *Tunable Variables* через пробел впечатываем настраиваемые коэффициенты $K_p, K_i. K_d$.

6. Запускаем процесс оптимизации, нажав кнопку *Start*. Вначале строится кривая переходного процесса с начальными значениями коэффициентов. Затем программа оптимизации продолжает итерационные шаги, пока кривая не впишется в заданный коридор. Соответствующие этой кривой значения коэффициентов ПИД–регулятора можно посмотреть в командной строке *Matlab*, набрав K_p и *Enter*, K_i и *Enter* и т. д. Получим $K_p = 2.0758$; $K_i = 0.081$; $K_d = 11.2424$.

Полученные значения существенно отличаются от начальных. Рассмотренный метод с использованием *Simulink* удобен для настройки ПИД–регуляторов.

Достаточно широко на практике используются ПИ-регуляторы, реализующие пропорционально-интегральный закон управления. Они настраиваются как и ПИД-регуляторы, с учетом того, что у ПИ-регулятора всегда коэффициент $K_d = 0$.

2 Порядок выполнения лабораторной работы

1. Разработать структурную схему САУ с ПИД-регулятором в соответствии с рис. 3.2.

2. Задать численные значения параметров звеньев структурной схемы (кроме ПИД-регулятора) в соответствии с приведенным в разделе 3 описанием. Выбрать из таблицы 1 значения $W_{OY}(s)$ для варианта №1.

3. Для переходного процесса задать следующие параметры:

- наибольшее перерегулирование $\sigma \leq 20\%$;

- длительность переходного процесса $t nn^{\leq} 30 c$;

- установившаяся ошибка $\varepsilon_{VCM} \leq 5\%$.

4. По приведенной в разделе 2 методике Зиглера-Николса осуществить настройку коэффициентов ПИД-регулятора.

5. Сравнить полученные результаты с начальными.

6. Выбрать из таблицы 1 значения *w*_{OV} (*s*) для варианта № 2. Вы-полнить п.п. 3—5.

Значения пе	редаточных функций <i>w_{OV}</i> (му Таблица 1		
	Вариант №1	Вариант №2		
Бригада №1	1	1		
Бртада №1	$\overline{40s^3 + 12s^2 + 7s + 1}$	$0.4s^3 + 12s^2 + 0.7s + 1$		
Бригала №2	1	1		
Бригада №2	$\overline{34s^3 + 12s^2 + 2s + 1}$	$0.6s^3 + s^2 + 0.3s + 1$		
Бригада №3	1	1		
вригада л≌э	$\overline{0.8s^3 + 0.02s^2 + s + 1}$	$52s^3 + 9s^2 + 9s + 1$		
Бригада №1	1	1		
Бршада л≞+	$\overline{15s^3 + 1.2s^2 + s + 1}$	s^3_+ 0.7 s^2_+ 0.06 s + 1		
Бригада Моб	1	1		
вригада №3	$\overline{22s^3 + 5s^2 + 2s + 1}$	$0.9s^3 + s^2 + 0.3s + 1$		
	1	1		
Бригада №6	$0.6s^3 + s^2 + 0.3s + 1$	$71s^3 + 9s^2 + s + 1$		

3 Требования к выполнению отчета

Отчет по лабораторной работе выполняется в отдельной тетради и должен содержать:

1. Название лабораторной работы, ее цель, задачи.

2. Схемы, выполняемые в Simulink.

3. Полученные в результате моделирования осциллограммы.

4. Полученные значения настраиваемых коэффициентов с необходимыми комментариями.

4 Вопросы для самопроверки

1. Что необходимо иметь для настройки промышленных регуляторов?

2. Какой закон управления реализуют ПИД-регуляторы?

3. За счет чего ПИД-регулятор обеспечивает устойчивость и минимизирует статическую и динамическую погрешности САУ?

4. В чем заключается настройка ПИД-регуляторов?

5. В чем заключается методика настройки Зиглера-Николса?

Лабораторная работа № 4

Экспериментальное исследование качества позиционной системы автоматического управления на базе двигателя постоянного тока с независимым возбуждением

Цель работы — получение практических навыков в определении качественных показателей систем управления с использованием переходных характеристик.

1 Сведения из теории [1, 5]

Один из этапов проектирования систем управления — их экспериментальное исследование с целью определения реальных показателей качества.

Исчерпывающее представление о качестве системы автоматического управления (САУ) дает переходная характеристика. Переходная характеристика h(t) — это изменение выходного сигнала y(t) при подаче на вход единичного ступенчатого воздействия x(t) 1(t) при-нулевых начальных условиях (то есть для системы, которая в начальном состоянии находится в покое $y(0) \neq (0) 0$. На

рис. 4.1 а, б представлены возможные переходные характеристики САУ.



Рис. 4.1 а), б). Переходные характеристики САУ

Кривые 1 (апериодический процесс) и 2 (колебательный процесс) характерны для практически неприемлемых САУ с большими временами переходных процессов $t_{1\Pi\Pi}$ и $t_{2\Pi\Pi}$. Наибольшим быстродействием (меньшим временем $t_{\Pi\Pi}$) обладают САУ с переходной характеристикой, представленной на рис. 4.1 б. По этой кривой определяются численные значения следующих показателей качества системы управления:

1. Время переходного процесса t $_{\Pi\Pi}$. Оно характеризует быстродействие системы и определяется как интервал времени от начала переходного процесса до момента, когда отклонение выходной величины от ее установившегося значения становится не более $\pm 5\%$, то есть ошибка регулирования

$$\{\delta_{PE\Gamma}(t) \neq \frac{|y(t)-y_{VCT}|}{y_{VCT}} \times 100\%\} \leq 5\%$$

2. Максимальное перерегулирование объределяет наиболь-ший «выбег» выходной величины относительно установившегося значения в переходном периоде. Оно определяется относительно установившегося значения в %.

$$\sigma = \frac{A1}{y} \times 100\%$$

3. Статическая ошибка ε *уст*, определяющая точность системы ε *уст* = lim [*y*(*t*) - *ууст*]
t $\rightarrow \infty$

4. Время максимального перерегулирования t_m , при котором выходная величина y(t) = h(t) достигает наибольшего значения в переходном процессе.

5. Число перерегулирований N в интервале $0 \notin t \leq_{\Pi\Pi}$. Число перерегулирований определяется числом выбросов, при которых кривая h(t) располагается ниже значения y_{YCT} . Для кривой h(t), представленной на рис. 4.16, $N \rightrightarrows$. Для практически приемлемых САУ $N \ge 4$

6. Показатель колебательности *М* переходного процесса. Это — отношение соседних максимумов в процентах (%.)

$$M = \frac{A1}{A2} \times 100\%$$

а) при 0 *М* 100% — система неустойчива, так как имеем ре-жим незатухающих колебаний с возрастающей амплитудой;

б) при *М* 100% — имеем режим автоколебаний с постоянной амплитудой;

в) при 100% « $M < \infty$ система устойчива, так как. это режим затухающих колебаний; устойчивая САУ практически приемлема, если $M = 100\% \div 150\%$

г) при $M = \infty$ — имеем апериодический режим (кривая 1 на рис. 4.1 а).

2 Схема лабораторного стенда

На рис. 4.2 представлена схема лабораторного стенда для экспериментального определения переходных характеристик s(t) = f(t), где s — величина линейного перемещения объекта управления ОУ в единицах отсчета цифрового датчика положения (ЦДП).



Рис. 4.2. Схема лабораторного стенда

САУ обеспечивает линейное перемещение ОУ на заданную длину с требуемой точностью, определяемой точностью ЦДП, являющегося звеном главной отрицательной обратной связи по положению (используется датчик типа ЛИР-120А). Перемещение ОУ осуществляется передачей винт-гайка (ПВГ) качения, которая приводится в движение двигателем М 1 (двигатель постоянного тока независимого возбуждения (ДПТНВ) типа СЛ621). Для создания регулируемого момента сопротивления на валу двигателя используется аналогичный двигатель М 2, работающий в генераторном режиме [6]. Тахогенератор (ТГ) постоянного тока типа ТГП-1 является звеном отрицательной обратной связи по скорости.

Цифровой сигнал с датчика положения вводится в процессор через блок *I/O* модуля *NI USB-6008* [7], а аналоговый сигнал с выхода ТГ — через АЦП того же модуля. Задающий код (соответствующий требуемой величине перемещения ОУ) подается с выхода процессора через ЦАП модуля *NI USB-6008* на вход силового преобразователя БОТ [8]. Кроме того, на вход обратной связи БОТ подается сигнал непосредственно с выхода ТГ.

Изначально задающий код изменяется скачком и позиционирование ОУ осуществляется с перерегулированием (кривая на рис. 4.1 б). Построение графика s(t) = f(t) выполняется средствами *LabView* [9].

3 Порядок выполнения работы

1. Перед выполнением работы пройти инструктаж по технике безопасности.

2. Проверить отсутствие напряжения питания силового преобразователя БОТ. Силовой автомат А1 в настенном шкафу должен быть выключен. При выключенном А1 собрать схему в соответствии с рис. 4.2.

3. Определить переходную характеристику s(t) = f(t) для вели-

чины перемещения ОУ $s = 50T \cdot m_{MM}$ при тормозящем токе в якоре M2, равном 0 A (дополнительный момент сопротивления $MT = 0^{\circ}$).

а) ОУ вручную переместить в крайнее левое положение (рис. 4.2).

б) Включить питание БОТ автоматом А1 в настенном шкафу. При этом загорается зеленый светодиод «Вкл» на индикаторе БОТ.

в) В поле рабочей программы «Использование датчиков» выбрать датчик ДП (датчик положения ЛИР-120А).

г) В рабочем окне программы [7] задать максимальное время *t*_{max} =1 *c* отработки входного воздействия. д) С помощью графического интерфейса программы задать ис-

ходное значение задающего кода $s_{3a\partial} = \frac{s [MM]}{\delta_{\partial am}[MM]}$, где $\delta_{\partial am} = \frac{\delta_{\partial am}}{\delta_{\partial am}[MM]}$

е) Нажать кнопку *Start* графического интерфейса.

4. Из полученного в окне рабочей программы графика занести в Таблицу 1 измеряемые значения.

5. Получившийся в окне рабочей программы график перерисовать в рабочую тетрадь, соблюдая масштаб.

6. Выключить силовой автомат А1 в настенном шкафу и повторить пункты 4—11 при *s* ±00 *мм* и *s* 150 *мм*.

7. Определить переходную характеристику s(t) = f(t) для величины перемещения ОУ s = 0 мм $M_{c \ don} \neq 0$. Величина $M_{c \ don}$ изменяется с помощью R_H (рис. 4.2) [6]. Далее выполнить пункты

8. Выключить силовой автомат А₁ в настенном шкафу.

Параметры качества переходного процесса

Таблица 1

	- r		1	/ \	1	1				1
Длина пере- меше-	1 V1 C	Пе лиј	сререн рован $\sigma, \%$	гу- ие	t _{nn} ,	ε,%	t_m ,	Пок леба	азате: ателы М, 9	пь ко- ности ⁄6
ния с		Δ	$\mathbf{Y}_{\mathbf{V}}$	G	C	%	C			
		1	Ту	0, 0/	U		U	A1	A2	M, %
MM		1	СТ	70						
50	0									
100	0									
150	0									
50	≠0									
100	<i>≠</i> 0									
150	≠ 0									

4 Требования к выполнению отчета

Отчет по лабораторной работе представляется каждым студентом и должен содержать:

1. Название лабораторной работы, ее цель.

2. Схему лабораторной работы, собираемую на стенде.

3. Шесть графиков переходных процессов, выполненных на миллиметровой бумаге.

4. Таблицу с измеренными по графикам и вычисленными значениями параметров качества переходного процесса.

5. Выводы о зависимости численных значений показателей качества переходного процесса от длины перемещений и величин моментов сопротивления.

5 Вопросы для самопроверки

1. Виды кривых переходных процессов позиционных САУ.

2 Показатели качества переходных процессов и способы их определения (вычисления).

3. Показатели качества практически приемлемой САУ.

Библиографический список

1. Гудвин, Г. К. Проектирование систем управления / Г. К. Гудвин, С. Ф. Гребе, М. Э. Сальгадо ; пер. с англ. А. М. Епанешникова. — М. : БИНОМ. Лаборатория знаний, 2004. — 911 с.

2. Черных, И. В. *SIMULINK* — среда создания инженерных приложений / И. В. Черных. — М. : Диалог-МИФИ, 2004. — 492 с.

3. Дьяконов, В. П. МАТLAB 6.5 SP1/7.0 + Simulink 5/6[®]. Основы применения / В. П. Дьяконов. — М. : СОЛОН-Пресс, 2005 — 800 с.

4. Олссон, Г. И Пиани Дж. Цифровые системы автоматизации и

управления / Г. Олссон, Дж. Пиани — СПб. : Невский Диалект, 2001. — 557 с.

5. Ерофеев, А. А. Теория автоматического управления : учебник для вузов / А. А. Ерофеев. — 2-е изд., перераб. и доп. — СПб.: Политехника, 2005. — 302 с.

6. Леонов, А. П. Лабораторные работы по дисциплине «Электрические машины и электропривод» : учебно-методическое посо-бие / А. П. Леонов. — Дубна : Междунар. ун-т природы, о-ва и челове-ка «Дубна», 2010. — 34 с.

7. User guide and specifications NI USB-6008/6009. — \bigcirc National Instruments Corporation, 2008. — 31 p.

8. Блоки однофазные тиристорные серии БОТ. Руководство по эксплуатации. — Чебоксары : ООО «ЭЛПРИ», 2002. — 51 с.

9. Тревис, Дж. LabView для всех / Джеффри Тревис Пер. с англ. Н. А. Клушин. — М. ДМК Пресс : Прибор Комплекс, 2005. — 544 с.

Оглавление

Введение 1	
Лабораторная работа № 1	
Лабораторная работа № 2	
Лабораторная работа № 3	
Лабораторная работа № 4	
Библиографический список	