

Лабораторная работа № 2

Моделирование систем управления в пакете SIMULINK

Цели работы

Освоение методов моделирования линейных систем в пакете SIMULINK.

Задачи работы

- научиться строить и редактировать модели систем управления в пакете SIMULINK
- научиться изменять параметры блоков
- научиться строить переходные процессы
- научиться оформлять результаты моделирования
- изучить метод компенсации постоянных возмущений с помощью ПИД-регулятора

(краткие теоретические сведения)

Создание моделей в SIMULINK

Пакет SIMULINK предназначен для моделирования систем. Вся модель строится из блоков, имеющих входы и выходы. Существует библиотека стандартных блоков, кроме того, можно создавать свои собственные блоки любой сложности. Существует две группы специальных устройств – источники сигналов (**Sources**) и устройства вывода (**Sinks**).

Блоки имеют названия. Для того, чтобы изменить название, надо щелкнуть по нему ЛКМ и отредактировать текст¹.

Каждый блок имеет свои настраиваемые свойства. Для их изменения надо дважды щелкнуть на блоке и изменить нужные значения в диалоговом окне.

Для того, чтобы повернуть блок на 90 градусов, надо выделить его и нажать клавиши **Ctrl+R**. Комбинация **Ctrl+I** позволяет выполнить зеркальное отражение входов и выходов.

Верхнее меню **Format** предназначено для изменения оформления выделенного блока. Также для этой цели можно использовать контекстное меню ПКМ – **Format**. Для выделенного блока можно изменить цвет текста и линий (**Foreground color**), цвет фона (**Background color**), вывести тень (**Show drop shadow**), переместить название на другую сторону (**Flip name**).

Для выделения одного блока или соединительной линии надо щелкнуть ЛКМ по нужному элементу. Для того, чтобы выделить несколько блоков, надо «обвести» их при нажатой ЛКМ. Клавиша **Delete** удаляют выделенную часть. Чтобы скопировать блок (или выделенную часть), надо перетащить его при нажатой *правой* кнопке мыши (ПКМ).

¹ В некоторых версиях МАТЛАБ существуют проблемы с русской буквой «Я», поэтому рекомендуется использовать английские названия блоков.

Блоки соединяются линиями связи, по которым распространяются сигналы. Для того, чтобы соединить блоки, надо щелкнуть ЛКМ по источнику сигнала и затем, при нажатой клавише **Ctrl**, по блоку-приемнику. Можно также протянуть мышкой линию связи между нужными выходом и входом.

Чтобы подать один сигнал на два блока (сделать «развилку»), надо сначала создать одну линию обычным способом. Чтобы провести вторую линию, следует нажать *правую* кнопку мыши на линии в точке развилки и протащить линию ко второму блоку.

Модель можно скопировать в буфер обмена в виде растрового рисунка. Для этого в окне модели надо выбрать в верхнем меню пункт **Edit – Copy model to clipboard**. Предварительно лучше уменьшить размеры окна до минимальных, чтобы не было белых полей.

Для того, чтобы запустить моделирование, надо щелкнуть ЛКМ по кнопке ► на панели инструментов. Эта же кнопка позволяет остановить моделирование при необходимости.

Параметры моделирования (метод интегрирования, обработка ошибок) устанавливаются с помощью окна **Simulation – Parameters**. Самые важные параметры – это время моделирования (**Stop time**) и метод численного интегрирования уравнений (**Solver options**).

Основные источники сигналов (Sources)



Constant – сигнал постоянной величины.



Step – ступенчатый сигнал, меняется время скачка (**Step Time**), начальное (**Initial Value**) и конечное значение (**Final Value**).



Ramp – линейно возрастающий сигнал с заданным наклоном (**Slope**). Можно задать также время начала изменения сигнала (**Start Time**) и начальное значение (**Initial Value**).



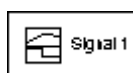
Pulse Generator – генератор прямоугольных импульсов, задаются амплитуда (**Amplitude**), период (**Period**), ширина (**Pulse Width**, в процентах от периода), фаза (**Phase Delay**).



Repeating Sequence – последовательность импульсов, их форма задается в виде пар чисел (время; величина сигнала)



Sine Wave – синусоидальный сигнал, задается амплитуда (**Amplitude**), частота (**Frequency**), фаза (**Phase**) и среднее значение (**Bias**).



Signal Builder – построитель сигналов, позволяющий задавать форму сигнала, перетаскивая мышью опорные точки.



Random Number – случайные числа с нормальным (гауссовым) распределением. Можно задать среднее значение (**Mean Value**), дисперсию (**Variance**), период изменения сигнала (**Sample Time**).

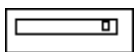


Uniform Random Number – случайные числа с равномерным распределением в заданном интервале от **Minimum** до **Maximum**.



Band Limited White Noise – случайный сигнал, ограниченный по полосе белый шум (имеющий равномерный спектр до некоторой частоты). Блок используется как источник белого шума для моделей непрерывных систем. Задается интенсивность (**Noise Power**) и интервал дискретизации (**Sample Time**), в течение которого удерживается постоянное значение сигнала. Чем меньше интервал, тем точнее моделирование, однако больше вычислительные затраты.

Основные устройства вывода (Sinks)



Display – цифровой дисплей, показывает изменение входного сигнала в цифровом виде.



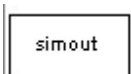
Scope – осциллограф, показывает изменение сигнала в виде графика, позволяет передавать данные в рабочую область MATLAB для последующей обработки и оформления.



XY Graph – графопостроитель, строит график одного сигнала в функции другого.



To File – блок сохранения данных в файле. Блок записывает данные, поступающие на его вход, в файл.

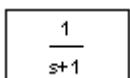


To Workspace – блок сохранения данных в рабочей области. Блок записывает данные, поступающие на его вход, в рабочую область MATLAB



Terminator – конечный приемник. Блок используется для подачи сигнала с неиспользуемого выхода другого блока

Линейные системы (Continuous)



Transfer Fcn – передаточная функция, в параметрах задаются числитель (**Numerator**) и знаменатель (**Denominator**) в виде полиномов.

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

State Space – модель в пространстве состояний, в параметрах задается четверка матриц, определяющих модель, и начальные условия для вектора состояния (**Initial conditions**).



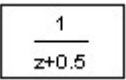
$$\frac{(s-1)}{s(s+1)}$$

Zero-Pole – модель в форме «нули-полюса», в параметрах задаются массивы нулей (**Zeros**), полюсов (**Poles**), а также коэффициент усиления (**Gain**).



$$\frac{1}{s}$$

Integrator – интегратор с возможностью установки начальных условий (**Initial condition**), а также пределов насыщения (**Lower saturation limit** и **Upper saturation limit**). Когда сигнал выхода выходит за границы, определяемые этими пределами, интегрирование прекращается.

Библиотека Discrete – дискретные блоки




Изображение	Название		Описание
	Unit Delay	Блок единичной дискретной задержки	Выполняет задержку входного сигнала на один шаг модельного времени
	Discrete-Time Integrator	Блок дискретного интегратора	Блок используется для выполнения операции интегрирования в дискретных системах
	Discrete Transfer Fcn	Дискретная передаточная функция	Блок Discrete Transfer Fcn задает дискретную передаточную функцию в виде отношения полиномов

Библиотека Math – блоки математических операций


Изображение	Название		Описание
	Sum	Блок вычисления суммы	Выполняет вычисление суммы текущих значений сигналов. Параметр List of signs задает количество входов, их знаки («+» для сложения и «-» для вычитания). Промежутки между входами (обозначаются знаком).
	Product	Блок умножения	Выполняет вычисление произведения текущих



			значений сигналов
	Sign	Блок определения знака сигнала	Определяет знак входного сигнала
	Gain и Matrix Gain	Усилители	Выполняют умножение входного сигнала на постоянный коэффициент
	Math Function	Блок вычисления математических функций	Выполняет вычисление математической функции
	Trigonometric Function	Блок вычисления тригонометрических функций	Выполняет вычисление тригонометрической функции
	MinMax	Блок определения минимального или максимального значения	Определяет максимальное или минимальное значение
	Relational Operator	Блок вычисления операции отношения	Блок сравнивает текущие значения входных сигналов
	Logical Operation	Блок логических операций	Реализует одну из базовых логических операций

Библиотека Look-Up Tables - Блоки таблиц

Изображение	Название		Описание
	Look-Up Table	Блок одномерной таблицы	Задаёт в табличной форме функцию одной переменной.
	Look-Up Table(2D)	Блок двумерной таблицы	Задаёт в табличной форме функцию двух переменных.
	Look-Up Table (n-D)	Блок многомерной таблицы	Задаёт в табличной форме функцию многих переменных.

Библиотека Signal&Routings - блоки преобразования сигналов

Изображение	Название		Описание
	Bus creator (Mux)	Блок шинного формирователя. мультиплексор	Формирует шину из сигналов различных типов, т.е. объединяет несколько сигналов в один «жгут» (векторный сигнал), в параметрах задается число входов (Number of Inputs).


	Manual switch	Блок ручного переключателя	Выполняет переключение входных сигналов по команде пользователя. Позволяет двойным щелчком переключать выход на один из двух входных сигналов
	Bus selector (Demux)	Блок шинного селектора (демультиплексор)	Выделяет из шины требуемые сигналы. Позволяет «разбить» векторный сигнал на несколько скалярных, в параметрах задается число выходов (Number of Outputs).


Таким образом, стандартными средствами Simulink можно создавать довольно сложные гибридные модели. Возникающие неудобства пользователя связаны скорее не с гибридной моделью, а с неудобством вообще набора сложных систем уравнений из примитивных блоков. Общим принципом гибридного моделирования в Simulink является использование готовых гибридных блоков или переключение заранее заготовленных альтернативных участков блок-схем. Ясно, что методом переключения ветвей блок-схемы принципиально невозможно моделировать системы с динамически изменяемой структурой. Возникают значительные трудности, связанные с описанием мгновенных действий при обработке дискретного события и тем более при описании цепочки дискретных событий во временной щели. (решения типа сброса интегратора следует признать все же искусственными). Кроме того, запутанные переключательные схемы чрезвычайно сложны для понимания.

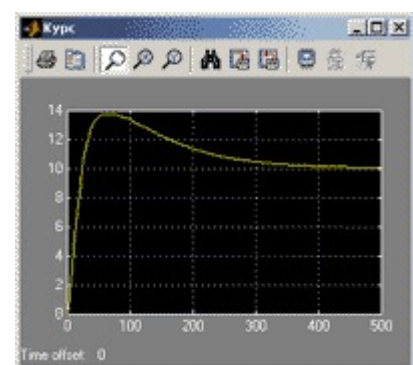
Для того, чтобы помочь пользователю преодолеть эти трудности, в последние версии Simulink введена специальная подсистема STATEFLOW, позволяющая создавать специальные дискретные блок. Блок «STATEFLOW» оформляется как подсистема и через свои входы и выходы может взаимодействовать с обычной блок-схемой. Безусловно, подсистема STATEFLOW существенно облегчает разработку нестандартных дискретных подсистем. Карта состояний позволяет легко задавать любую сложную логику возникновения дискретных событий и их обработки, а также имеет очень наглядное визуальное представление.


Блок Scope

В окне блока **Scope** изображается график изменения входного сигнала. Если вход соединен с выходом мультиплексора, сразу строится несколько графиков (по размерности входного «жгута»).

По умолчанию на оси ординат используется диапазон от -5 до 5. Если этот вариант не подходит, выбрать масштаб автоматически (так, чтобы весь график был виден) можно с помощью кнопки .

Соседняя кнопка  сохраняет эти настройки для следующих запусков.



Кнопка  открывает окно настроек, причем наиболее важные данные содержатся на вкладке **Data history**. Если не сбросить флажок **Limit data points**, в памяти будет сохраняться только заданное число точек графика, то есть, при большом времени моделирования начало графика будет потеряно.

Отметив на этой же странице флажок **Save data to workspace** можно сразу передать результаты моделирования в рабочую область MATLAB для того, чтобы их можно было дальше обрабатывать, выводить на графики и сохранять в файле. Поле **Variable name** задает имя переменной в рабочей области, в которой сохраняются данные. В простейшем случае выбирается формат **Array** (в списке **Format**). Это означает, что данные будут сохраняться в массиве из нескольких столбцов (первый столбец – время, второй – первый сигнал, третий – второй сигнал и т.д., по порядку входов мультимплексора).

Оформление графиков

Для создания нового окна для рисунка в MATLAB используется команда

```
>> figure(1);
```

Вместо единицы можно ставить любой номер рисунка. Если рисунок с таким номером уже есть, он становится активным и выводится на первый план. Если такого рисунка нет, он создается и становится активным.

В MATLAB есть возможность строить несколько графиков на одном рисунке. Иначе говоря, рисунок можно разбить на «клетки», в каждой из которых строится отдельный график. Для этого надо сделать активным нужный рисунок и применить команду

```
>> subplot(2, 1, 1);
```

Первое число в команде **subplot** показывает количество «строк» в такой матрице, второе – количество столбцов, третье – какой по счету график сделать активным (считая по строкам, справа налево и сверху вниз). Все дальнейшие команды (**plot**, **title**, **xlabel**, **ylabel**, **legend** и др.) относятся к этому «подграфику».

В командах можно передавать в качестве аргументов не целые массивы, а их части. Например, по команде

```
>> plot(x(1:20), y(11:30));
```

строится график, на котором по оси абсцисс откладываются значения элементов массива **x** с номерами от 1 до 20, а по оси ординат – соответствующие им значения из массива **y** с номерами от 11 до 30.

Двоеточие означает «все строки» или «все столбцы». Например, по команде

```
>> plot(x(:,1), x(:,2));
```

строится зависимость между первым и вторым столбцами массива **x** (здесь двоеточие вместо первого индекса обозначает «все строки»).

С помощью команды **plot** (а также и других подобных – **semilogx**, **semilogy**, **loglog**) можно строить несколько линий на одном графике. Для этого среди аргументов перечисляются пары массивов:

```
>> plot(x, y, v, z);
```

Первая линия будет показывать зависимость **y** от **x**, а вторая – зависимость **z** от **v**. массивы в каждой паре должны быть одинаковой длины. При желании можно указать цвета для каждой линии, Например,

```
>> plot(x, y, 'b', v, z, 'g');
```

Первая линия (зависимость **y** от **x**) будет синей, вторая (зависимость **z** от **v**) – зеленой. Можно использовать следующие цвета

b	синий (blue)
g	зеленый (green)
r	красный (red)
c	голубой (cyan)
m	фиолетовый (magenta)
y	желтый (yellow)
k	черный (black)

По умолчанию первая линия – синяя, вторая – зеленая и т.д. в порядке перечисления цветов в списке. Дополнительно можно указать тип линии

-	сплошная
:	точечная
-.	штрих-пунктирная
--	штриховая

Например,

```
>> plot(x, y, 'b:', v, z, 'g--');
```

Первая линия – точечная синего цвета, вторая – штриховая зеленого цвета. По умолчанию все линии сплошные.

Для оформления графика также используются команды

title	заголовок графика
xlabel	название оси абсцисс
ylabel	название оси ординат

У всех этих команд обязателен один аргумент – текст в апострофах.

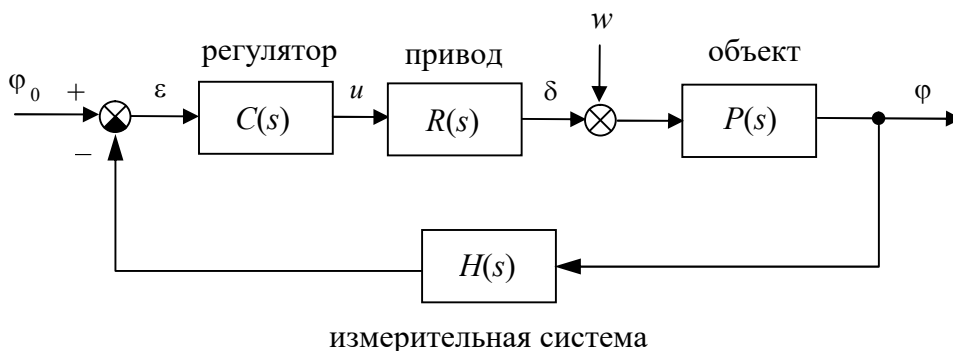
Команда **legend** служит для вывода легенды графика. Легенда нужна, если на графике есть несколько линий и надо показать, что обозначает каждая из них.

Параметрами команды **legend** являются символьные строки, их должно быть столько, сколько построено линий.

В надписях можно использовать некоторые команды системы TeX². Например, греческие буквы записываются в виде «\alpha», «\beta» и т.д. Верхний индекс (степень) обозначается знаком «^», Например, a^2 запишется как «a^2». Для обозначения индекса используют нижнее подчеркивание, например, a_{22} кодируется как «a_{22}».

Компенсация постоянных возмущений

На любое судно в реальных условиях действуют *возмущающие силы*, вызванные ветром, морским волнением и другими причинами. Некоторые из них (например, влияние ветра) содержат постоянную составляющую, т.е., их среднее значение не равно нулю. Тем не менее, система управления должна поддерживать заданный курс судна даже в таких условиях. Возмущающие силы и моменты приложены непосредственно к входу объекта управления, т.е., структурная схема имеет такой вид:



Подавление возмущений (обозначенных на схеме через w) определяется передаточной функцией системы по возмущению, т.е., передаточной функцией от входа w к выходу φ :

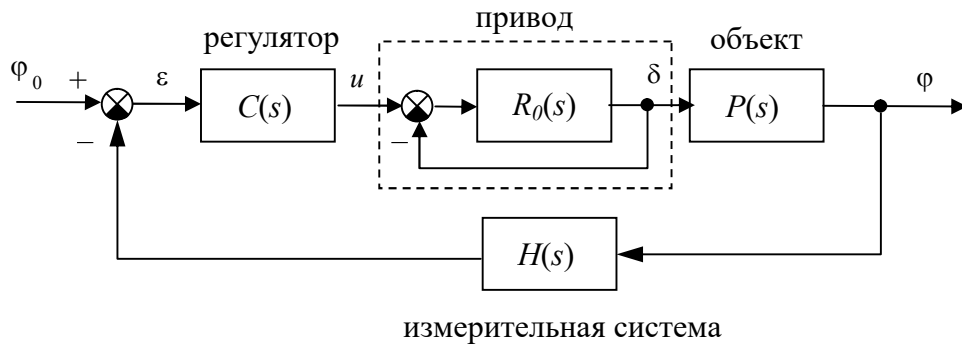
$$W_w(s) = \frac{P(s)}{1 + R(s)C(s)H(s)P(s)}.$$

Если она содержит нуль в точке $s=0$, соответствующая АЧХ равна нулю на нулевой частоте, т.е., постоянные возмущения в установившемся режиме компенсируются полностью. Для этого требуется, чтобы интегратор входил в модель привода, обратной связи или регулятора. Таким образом, если регулятор содержит интегральный канал (И-канал), в системе нет статической ошибки при постоянном возмущении w .

² TeX – лучшая в мире система подготовки текстов с математическими формулами, разработанная математиком и программистом Дональдом Кнутом (автором многотомного издания «Искусство программирования для ЭВМ»). Чаше всего используются так называемые макропакеты (надстройки), расширяющие возможности ядра TeX, например LaTeX или AMSTeX.

Описание системы

В работе рассматривается система управления судном по курсу. Ее структурная схема показана на рисунке.



Структурная схема системы стабилизации судна на курсе

Линейная математическая модель, описывающая рыскание судна, имеет вид

$$\dot{\varphi} = \omega_y$$

$$\dot{\omega}_y = -\frac{1}{T_s} \omega_y + \frac{K}{T_s} \delta$$

где φ – угол рыскания (угол отклонения от заданного курса), ω_y – угловая скорость вращения вокруг вертикальной оси, δ – угол поворота вертикального руля относительно положения равновесия, T_s – постоянная времени, K – постоянный коэффициент, имеющий размерность *рад/сек*. Передаточная функция от угла поворота руля к углу рыскания запишется в виде

$$P(s) = \frac{K}{s(T_s s + 1)}.$$

Привод (рулевая машина) приближенно моделируется как интегрирующее звено, охваченное единичной отрицательной обратной связью, так что его передаточная функция равна

$$R(s) = \frac{1}{T_R s + 1}.$$

Для измерения угла рыскания используется гирокомпас, математическая модель которого записывается в виде апериодического звена первого порядка с передаточной функцией³

$$H(s) = \frac{1}{T_{oc} s + 1}.$$

Исследуются переходные процессы в системе при использовании ПД-регулятора

$$C(s) = K_c \left(1 + \frac{T_s s}{T_v s + 1} \right),$$



и ПИД-регулятора

$$C(s) = K_c \left(1 + \frac{T_s s}{T_v s + 1} \right) + \frac{1}{T_I s}.$$

³ Численные значения K , T_s , T_R и T_{oc} надо взять из таблицы в конце файла. Они должны совпадать с данными, которые использовались Вами в лабораторной работе № 2.

Ход работы.

Инструкция по выполнению работы

Этап выполнения задания	Команды MATLAB
1. Для запуска пакета SIMULINK щелкните по кнопке  в командном окне MATLAB или введите команду simulink в командной строке.	simulink
2. Создайте новую модель с помощью верхнего меню открывшегося окна <i>Simulink Library Browser</i> .	 Simulink Library Browser File – New – Model
3. Перетащите блок Transfer Fcn (передаточная функция) из окна <i>Simulink Library Browser</i> (группа Continuous) в окно модели и введите числитель и знаменатель передаточной функции модели судна.	Двойной щелчок на блоке <ul style="list-style-type: none"> • Numerator[K] • Denominator [Ts 1 0]
4. Дайте блоку название Судно .	ЛКМ на имени блока
5. Аналогично добавьте еще три блока типа Transfer Fcn , назовите их Привод , Регулятор и Гирокомпас , введите нужные параметры. Заметьте, что передаточная функция привода должна быть $R(s) = \frac{1}{T_R s + 1}$ с учетом внутренней обратной связи.	
6. Сохраните модель в своей папке под именем lab2.mdl ⁴ .	File – Save
7. Выделите блок Гирокомпас и разверните его в другую сторону.	Нажать Ctrl+I или дважды нажать Ctrl+R.
8. Сделайте, чтобы названия блоков Судно , Привод и Регулятор были над блоками.	ПКМ на блоке, Format - Flip name
9. Выберите цвет блоков на свой вкус.	ПКМ на блоке, Format – Background color
10. Перетащите в окно модели блок Sum из группы Math Operations и установите его слева от регулятора.	ЛКМ
11. Сделайте так, чтобы второй вход учитывался в сумме со знаком минус (отрицательная обратная связь).	Двойной щелчок на блоке, ввести +- в поле List of signs
12. Перетащите в окно модели блок Step из группы Sources и установите его слева от сумматора. Дайте ему имя Заданный курс .	

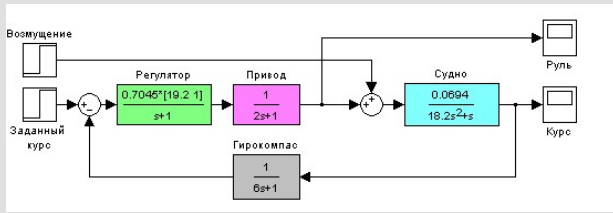
⁴ Все файлы моделей в пакете SIMULINK имеют расширение **.mdl**.

13. Установите время подачи сигнала 0 и величину сигнала 10 (исследуем поворот на 10 градусов).	Двойной щелчок на блоке, 0 в поле Step time 10 в поле Final value
14. Соедините все блоки нужным способом.	ЛКМ на источнике, удерживать Ctrl и ЛКМ на приемнике, или протащить ЛКМ от выхода одного блока к входу другого
15. Перетащите в окно модели два блока Scope (осциллограф) из группы Sinks и установите их в правой части. Назовите их Руль и Курс .	
16. Сделайте, чтобы на первый блок Scope поступал сигнал управления (угол поворота руля, после блока Привод), а на второй – сигнал выхода (курс судна). Сохраните модель.	Нажать ПКМ на линии в точке отбора сигнала, затем, не отпуская ПКМ, тащить линию к входу блока.
17. Уменьшите окно до минимального размера, при котором видны все элементы, и скопируйте модель в буфер обмена. Затем вставьте ее из буфера обмена в отчет.	Edit – Copy model to clipboard
18. Установите время моделирования 100 секунд.	Simulation – Simulation parameters 100 в поле Stop time
19. Выполните моделирование.	ЛКМ по кнопке ▶
20. Посмотрите результаты моделирования, открыв окна для блоков Курс и Руль .	Двойной щелчок по блоку
21. Настройте масштаб по осям в окнах обоих блоков,	ЛКМ по кнопке 🏠 – установить оптимальный масштаб
22. Сохраните настройки,	ЛКМ по кнопке 📄
23. Сделайте так, чтобы результаты моделирования передавались с обоих блоков Scope в рабочую область MATLAB в виде матриц, в которых первый столбец – время, а второй – сигнал (курс или угол поворота руля).	ЛКМ по кнопке 📄 вкладка Data history <input type="checkbox"/> Limit data points <input checked="" type="checkbox"/> Save data to workspace Variable name: phi (Курс) или delta (Руль) Format: Array
24. Выполните моделирование еще раз.	ЛКМ по кнопке ▶
25. Перейдите в командное окно MATLAB и создайте новое окно для графика. В одном окне будут построены две кривых на разных осях.	figure(1);

26. Разбейте окно на 2 части по вертикали и сделайте активным первый график. Первое число в команде subplot означает количество ячеек с графиками по вертикали, второе – по горизонтали, третье – номер ячейки, которую надо сделать активной ⁵ .	<code>subplot(2, 1, 1);</code>
27. Постройте график изменения курса. В команде plot сначала указывают массив абсцисс, затем – массив ординат. Двоеточие означает, что используются все строки.	<code>plot(phi(:,1),phi(:,2));</code>
28. Введите заголовок графика.	<code>title('Курс');</code>
29. Введите названия осей координат. Внутри апострофов для ввода греческих букв разрешается использовать команды LaTeX, Например, « <code>\phi</code> » означает греческую букву ϕ , а « <code>\delta</code> » – букву δ .	<code>xlabel('Время, сек');</code> <code>ylabel('phi, градусы');</code>
30. Аналогично постройте во второй ячейке график изменения угла поворота руля, используя данные из массива delta , полученного в результате моделирования.	<code>subplot(2, 1, 2);</code> <code>plot(delta(:,1),delta(:,2));</code> <code>title('Угол поворота руля');</code> <code>xlabel('Время, сек');</code> <code>ylabel('\delta, градусы');</code>
31. Скопируйте построенный график в отчет.	<code>print -dmeta</code>
32. Удалите в окне модели связь между приводом и объектом.	ЛКМ по линии, нажать Delete.
33. Добавьте еще один блок Sum из группы Math Operations и установите его на освободившееся место. Настройте расположение входов и выхода так, чтобы первый вход был в верхней части круга..	Двойной щелчок по блоку ++ в поле List of signs
34. Исследуем реакцию системы на постоянный сигнал, приложенный непосредственно к входу объекта. Он может моделировать какое-то постоянное возмущающее воздействие, например, влияние ветра.	
35. Скопируйте блок Заданный курс , перетащив его правой кнопкой мыши, и установите для него величину скачка 2 градуса. Дайте ему название Возмущение . Подключите его выход к новому сумматору. Достройте нужные соединительные линии.	Перетаскивание ПКМ. Двойной щелчок по блоку 2 в поле Final Value Двойной щелчок по имени

⁵ При вводе этой и следующих команд окно с графиком не появляется на экране. Чтобы увидеть изменения, надо вручную сделать его активным, щелкнув мышью на соответствующей кнопке в панели задач.

36. Скопируйте полученную модель в отчет.



Edit – Copy model to clipboard

37. Увеличьте время моделирования до 500 и выполните моделирование. Проверьте, вышло ли судно на заданный курс 10 градусов.

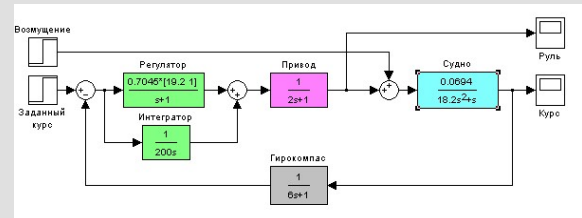
Simulation – Simulation parameters - Stop time
ЛКМ по кнопке ▶

38. Постройте передаточную функцию по возмущению замкнутой системы с ПД-регулятором. С ее помощью объясните результат, полученный на предыдущем шаге.

39. Перейдите в командное окно MATLAB и запомните результаты моделирования в новых массивах. Они понадобятся для того, чтобы сравнить исходный и скорректированный варианты системы.

phi0 = phi;
delta0 = delta;

40. Чтобы регулятор компенсировал постоянную составляющую возмущения, надо добавить в него интегральный канал. Таким образом, получается ПИД-регулятор. Подключите параллельно регулятору интегрирующее звено с передаточной функцией $\frac{1}{T_i s}$, $T_i = 200$ сек. Сохраните модель и скопируйте ее в отчет.



41. Выполните моделирование. Проверьте, вышло ли судно на заданный курс 10 градусов.

ЛКМ по кнопке ▶

42. Постройте передаточную функцию по возмущению замкнутой системы с ПИД-регулятором. С ее помощью объясните результат, полученный на предыдущем шаге.

43. Постройте в верхней части графика 2 кривых – переходные процессы по курсу для ПД- и ПИД-регуляторов. В команде **plot** можно перечислять несколько пар массивов – первая пара соответствует первому графику, вторая – второму и т.д. Три точки в конце строки означают перенос команды на следующую строку. Команда **legend** служит для вывода легенды – символьных строк, описывающих каждый из построенных графиков.

```
subplot(2, 1, 1);
plot(phi0(:,1), phi0(:,2),...
     phi(:,1), phi(:,2));
title('Курс');
xlabel('Время, сек');
ylabel('\phi, градусы');
legend('ПД-регулятор', ...
      'ПИД-регулятор');
```

44. Аналогично постройте в нижней части графика 2 кривых – изменение угла перекадки руля для ПД- и ПИД-регуляторов, используя данные из массивов **delta0** и **delta**.

Таблица коэффициентов

Вариант	T_s , сек	K , рад/сек	T_R , сек	T_{oc} , сек
1.	16.0	0.06	1	1
2.	16.2	0.07	2	2
3.	16.4	0.08	1	3
4.	16.6	0.07	2	4
5.	16.8	0.06	1	5
6.	17.0	0.07	2	6
7.	17.2	0.08	1	1
8.	17.4	0.07	2	2
9.	17.6	0.06	1	3
10.	17.8	0.07	2	4
11.	18.0	0.08	1	5
12.	18.2	0.09	2	6
13.	18.4	0.10	1	1
14.	18.6	0.09	2	2
15.	18.8	0.08	1	3
16.	19.0	0.07	2	4
17.	19.2	0.08	1	5
18.	19.4	0.09	2	6
19.	19.6	0.10	1	1
20.	18.2	0.0694	2	6

Контрольные вопросы к защите

1. Как найти передаточную функцию интегратора, охваченного обратной связью?
2. Как запустить пакет SIMULINK?
3. Что такое *Library Browser*?
4. Какое расширение имеют файлы – модели SIMULINK?
5. Как создать новую модель?
6. Как соединить два блока, имеющих соответственно свободный выход и свободный вход?
7. Как сделать, чтобы один и тот же сигнал поступал на несколько блоков?
8. Как передать результаты моделирования в рабочую область MATLAB? В каком виде они передаются?
9. Как удалить блок или связь между блоками?
10. Как определить нужные масштабы для осей координат в окнах **Scope** и запомнить их?
11. Как скопировать блок в окне модели?
12. Как изменить знаки арифметических действий в сумматоре?

13. Как скопировать изображение модели в документ *Microsoft Word*?
14. Как изменить время моделирования?
15. Как изменить название у блока?
16. Как сделать, чтобы название блока было с другой стороны?
17. Как изменить цвет фона блока? цвет надписи?
18. Как ввести параметры блока **Transfer Fcn** (передаточная функция)?
19. Как найти передаточную функцию системы по возмущению?
20. Почему при использовании ПД-регулятора система не компенсирует постоянное возмущение?
21. Как, зная статический коэффициент усиления по возмущению, определить установившееся отклонение от заданного курса?
22. Какими свойствами должна обладать передаточная функция по возмущению для того, чтобы постоянное возмущение полностью компенсировалось?
23. Какими свойствами должен обладать регулятор для того, чтобы постоянное возмущение полностью компенсировалось?
24. Какие преимущества дает использование интегрального канала в ПИД-регуляторе?
25. Почему порядок передаточной функции замкнутой системы по возмущению с ПИД-регулятором на 1 больше, чем для системы с ПД-регулятором?
26. Какие параметры принимает команда **subplot**?
27. Что означает двоеточие в записи **phi(:,1)**?
28. Как вывести на график заголовки и названия осей?
29. Как построить в одном окне два разных графика?
30. Как на одном графике построить несколько кривых?
31. Что такое легенда? Как вывести легенду на график?
32. Как выводить на графике буквы греческого алфавита?

