Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Оренбургский государственный университет»

Кафедра информатики

В.В. Извозчикова

МОДЕЛИРОВАНИЕ ДИНАМИКИ ПРОЦЕССОВ И СИСТЕМ ИНСТРУМЕНТАЛЬНЫМИ СРЕДСТВАМИ

Методические указания

Рекомендовано к изданию редакционно-издательским советом федерального государственного бюджетного образовательного учреждения высшего образования «Оренбургский государственный университет» для обучающихся по образовательной программе высшего образования по направлению подготовки 09.03.02 Информационные системы и технологии

Оренбург
2019

УДК 004.65(076)
ББК 32.973-018.2я7
И34

Рецензент – доцент, кандидат технических наук А.Н. Колобов

Извозчикова, В. В.
И34 Моделирование динамики процессов и систем инструментальными средствами: методические указания / В.В. Извозчикова - Оренбургский гос. ун.-т. – Оренбург: ОГУ, 2019

Методические указания предназначены для выполнения курсовых работ и расчетно-графических заданий, связанных с работой в инструментальном средстве Matlab/ Simulink, содержат краткое теоретическое описание по моделированию динамики процессов и систем и примеры работы в Matlab/ Simulink.

Методические указания предназначены для обучающихся по направлению подготовки 09.03.02 Информационные системы и технологии при изучении дисциплины «Инструментальные средства информационных систем»

УДК 004.65(076)
ББК 32.973-018.2я7

Содержание

Введение

Достижения в теории и практике моделирования процессов и систем, в современных условиях, связано со стремительным развитием вычислительной техники. Что казалось невозможным при решении многих задач моделировании еще несколько лет назад, сейчас легко реализуется на доступном инженерном уровне. Появление и развитие инженерных пакетов моделирования, таких как Matlab, Skylab, Labview, создало условия высокопроизводительного, объектно-ориентированного моделирования на современных компьютерах [1].

Задачи моделирования процессов и систем многообразны. Моделирование широко используется при инженерном проектировании и научных исследованиях: для решения технических и экономических задач, при исследованиях в экологии и социологии, в приборостроении и автоматизации управления.

Особенности применения моделирования в приборостроении связаны в первую очередь с технологическими достижениями в датчикостроении, теории измерений и обработки информации.

В области экономических задач применение моделирования дает эффективный инструмент для управления проектами и прогнозирования развития экономических процессов. Многие современные методы теории управления оказались эффективными при решении экономических задач и достаточно легко реализуемыми на математических моделях и постановке вычислительных экспериментов на компьютерной технике.

Развитие нейросетей, микросистемотехники, нанотехнологии внесло много существенно нового в методы моделирования процессов и систем, что дало также эффективный инструмент для предварительного решения задач проектирования в математическом виде на моделях и их численном исследовании на компьютерах.

Применение моделирования особенно эффективно при исследовании проектируемых систем с целью изучения и прогнозировании различных явлений и процессов в этих системах. Приближение к реальным условиям работы проектируемых систем осуществляется при стохастическом моделировании, когда к условиям моделирования добавляются случайные изменения параметров системы, возмущения и шумы измерений физических величин.

В приборостроении актуально моделирование задач управления, получения, передачи и преобразования информации. При этом современные модели везде для описания процессов и систем используют дифференциальные уравнения и линейные матричные преобразования.

Развитие современных методов моделирования создало предпосылки для создания и исследования высокоэффективных систем, которые, как правило, ориентированы на цифровые алгоритмы обработки информации, с применением современных микропроцессоров, нейрокомпьютеров, процессоров с нечеткой логикой и других современных технологических достижений.

1 Теоретическая часть

1.1 Применение пакета Simulink для моделирования динамики процессов и систем

Для проектирования и анализа различных динамических систем в настоящее время широко используются средства вычислительной техники, к которым относится пакет Simulink, входящий в состав пакетов расширений MATLAB [2]. Кроме того, пакет Simulink позволяет существенно удешевить и ускорить этап проектирования систем и устройств в механике, в электротехнике, радиотехнике, в биологии и химии, а также во многих других областях науки и техники

Simulink – интерактивный инструмент для моделирования, имитации и анализа динамических систем. Он дает возможность строить графические блок-диаграммы, имитировать динамические системы, исследовать работоспособность систем и совершенствовать проекты. Simulink полностью интегрирован с MATLAB, обеспечивая немедленный доступ к широкому спектру инструментов анализа и проектирования. Simulink также интегрируется с Stateflow для моделирования поведения, вызванного событиями. Эти преимущества делают Simulink наиболее популярным инструментом для проектирования систем управления и коммуникации, цифровой обработки и других приложений моделирования.

Simulink является самостоятельным инструментом MATLAB и при работе с ним совсем не требуется знать сам MATLAB и остальные его приложения. С другой стороны доступ к функциям MATLAB и другим его инструментам остается открытым и их можно использовать в Simulink [3].

Пакет Simulink служит для блочного моделирования различных систем и устройств. При моделировании пользователь на экране из библиотеки стандартных блоков создает модель устройства и осуществляет расчеты. Пользователь может выбирать метод решения дифференциальных уравнений, а также способ изменения модельного времени (с фиксированным или переменным шагом). В ходе моделирования имеется возможность следить за процессами, происходящими в системе. Для этого используются специальные устройства наблюдения, входящие в состав библиотеки Simulink. Результаты моделирования могут быть представлены в виде графиков или таблиц.

В пакете Simulink имеется возможность считывать значения переменных, находящихся в рабочем пространстве MATLAB (Workspase), и сохранять результаты расчетов в текстовом файле.

Simulink автоматически составляет и решает сложные системы алгебраических и дифференциальных уравнений, описывающих созданную модель.

Основным достоинством пакета Simulink является возможность, почти мгновенно менять математическое описание модели по мере ввода в нее новых блоков, даже в том случае, когда это изменяет порядок системы уравнений и ведет к существенному качественному изменению поведения системы.

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

Simulink – представитель визуально-ориентированного языка программирования. На всех этапах работы, особенно при подготовке схем моделей, пользователь вообще не имеет дела с программированием как с таковым. Нужная программа автоматически генерируется по мере ввода выбранных автоматически генерируется по мере ввода выбранных блоков, их соединений друг с другом и по мере задания параметров.

1.2 Интерфейс пакета Simulink

После открытия основного окна программы MATLAB нужно запустить программу Simulink. Это можно сделать одним из способов (рисунок 1.1) [3]:

– нажать на кнопку , расположенную на панели инструментов (в результате чего откроется окно обозревателя разделов библиотеки Simulink);

– выполнить команду File → New → Model (в результате чего откроется окно создания модели)

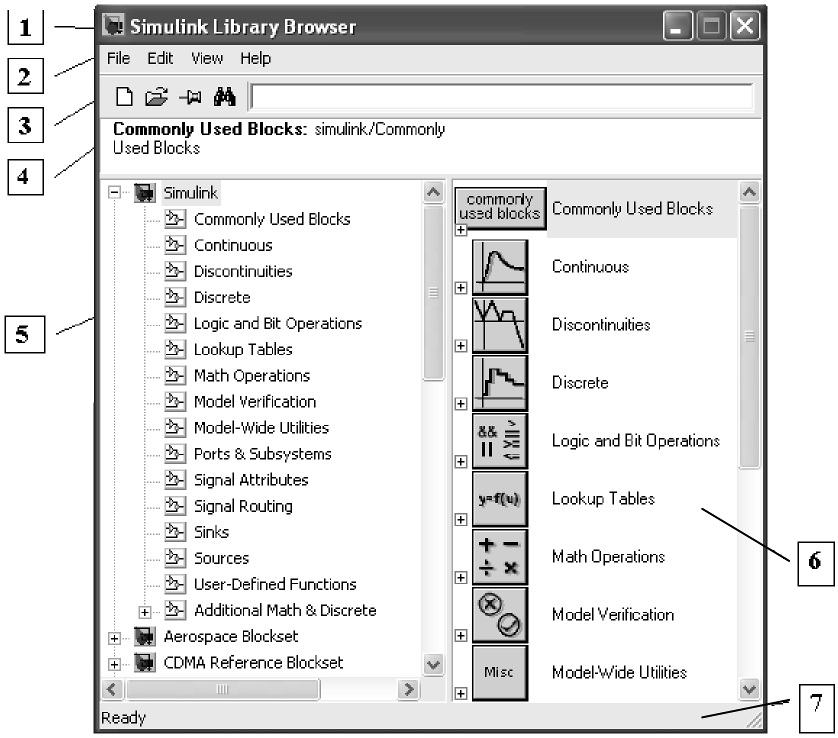


Рисунок 1.1 – Окно обозревателя разделов библиотеки Simulink

Окно обозревателя библиотеки блоков содержит следующие элементы.

1. Заголовок, с названием окна – Simulink Library Browser.

2. Меню обозревателя библиотек.

3. Панель инструментов, с кнопками наиболее часто используемых команд.

4. Окно комментария для вывода поясняющего сообщения о выбранном блоке.

5. Список разделов библиотеки, реализованный в виде дерева.

6. Окно содержимого раздела библиотеки (список вложенных разделов библиотеки или блоков).

7. Строка состояния, содержащая подсказку по выполняемому действию.

Библиотека Simulink содержит следующие основные разделы:

– «Commonly Used Blocks» – часто используемые блоки;

– «Continuous» – линейные компоненты;

– «Discontinuities» – прерыватели;

– «Discrete» – открытие окна с перечнем дискретных компонентов;

– «Logic and Bit Operations» – логические и битовые операторы;

– «Lookup Tables» – вспомогательные таблицы;

– «Math Operations» – математические операторы;

– «Functions & Table» – функции и таблицы;

– «Nonlinear» – нелинейные блоки;

– «Signals & Systems» – сигналы и системы;

– « Subsystems» – блоки подсистем;

– «Model Verification» – проверка модели;

– «Ports & Subsystems» – порты и подсистемы;

– «Sinks» – регистрирующие устройства;

– «Sources» – источники сигналов и воздействий;

– «User-Defined Function» – пользовательская функция.

Меню обозревателя библиотек содержит следующие пункты.

«File (Файл)» – работа с файлами библиотек.

«Edit (Редактирование)» – добавление блоков и их поиск (по названию).

«View (Вид)» – управление показом элементов интерфейса.

«Help (Справка)» –вывод окна справки по обозревателю библиотек.

Для работы с обозревателем можно также использовать кнопки на панели инструментов:

Рисунок 1.2 – Назначение кнопок на панели инструментов

Кнопки панели инструментов имеют следующее назначение.

1. Создать новую модель (открыть новое окно модели).

2. Открыть одну из существующих моделей.

3. Изменить свойства окна обозревателя. Данная кнопка позволяет установить режим отображения окна обозревателя "поверх всех окон”. Повторное нажатие отменяет такой режим.

4. Поиск блока по названию (по первым символам названия). После того как блок будет найден, в окне обозревателя откроется соответствующий раздел библиотеки, а блок будет выделен. Если же блок с таким названием отсутствует, то в окне комментария будет выведено сообщение Not found <имя блока> (Блок не найден).

1.3 Понятие динамических систем

Динамические системы - системы, под действием внешних и внутренних сил изменяющие во времени свои состояния. Представления о динамических системах возникли как обобщение понятия механической системы, поведение которой описывается законами динамики Ньютона. В современной науке понятие динамической системы охватывает системы практически любой природы: физические, химические, биологические, экономические, социальные и др. При этом системы характеризуются различной внутренней организацией -жестко-детерминированные, стохастические, нелинейные, системы с элементами самоорганизации, самоорганизующиеся.

Важнейшим свойством динамических систем является их устойчивость, т. е. сохранение системой своей базовой структуры и основных выполняемых функций в течение определенного времени и при относительно небольших и разнообразных внешних воздействиях и внутренних возмущениях. Устойчивость есть внутреннее свойство систем, а не результат внешнего воздействия. Представления же о развитии этих систем отражают такие изменения их структурной организации, которые ведут к более эффективному выполнению системой своих основных функций. Качественные перестройки систем анализируются в теории катастроф, которая рассматривается как ветвь общей теории динамических систем [4].

Развитие представлений о динамических системах связано с переходом к познанию все более сложных систем. При этом особую роль приобретает изучение динамики внутренних свойств систем. В случае механических систем действие внутренних факторов сводилось к силам инерции. По мере усложнения систем возрастает значение внутренних факторов. На первый план выходят проблемы изучения источников внутренней активности систем и природы их целенаправленного функционирования и поведения.

Математической моделью динамической системы принято называть совокупность математических символов, однозначно определяющих развитие процессов в системе, т.е. ее движение. При этом в зависимости от используемых символов различают аналитические и графоаналитические модели. Аналитические модели строятся с помощью буквенных символов, в то время как графоаналитические допускают применение графических обозначений.

В зависимости от типа сигналов различаются непрерывные и дискретные модели систем. В зависимости от используемых операторов - линейные и нелинейные, а также временные и частотные модели. К временным относятся модели, в которых аргументом является (непрерывное или дискретное) время. Это дифференциальные и разностные уравнения, записанные в явном виде или в операторной форме. Частотные модели предусматривают использование операторов, аргументом которых является частота соответствующего сигнала, т.е. операторы Лапласа, Фурье и т.д.

1.4 Временной ряд и его основные элементы

Временной ряд – это совокупность значений какого-либо показателя за несколько последовательных моментов или периодов времени. Каждый уровень временного ряда формируется под воздействием большого числа факторов, которые условно можно подразделить на три группы:

- факторы, формирующие тенденцию ряда;
- факторы, формирующие циклические колебания ряда;
- случайные факторы.

При различных сочетаниях в изучаемом процессе или явлении этих факторов зависимость уровней ряда от времени может принимать различные формы. Во-первых, большинство временных рядов экономических показателей имеют тенденцию, характеризующую долговременное совокупное воздействие множества факторов на динамику изучаемого показателя. Очевидно, что эти факторы, взятые в отдельности, могут оказывать разнонаправленное влияние на исследуемый показатель. Однако в совокупности они формируют его возрастающую или убывающую тенденцию.

Во-вторых, изучаемый показатель может быть подвержен циклическим колебаниям. Эти колебания могут носить сезонный характер, поскольку деятельность ряда отраслей экономики и сельского хозяйства зависит от времени года. При наличии больших массивов данных за длительные промежутки времени можно выявить циклические колебания, связанные с общей динамикой временного ряда.

Некоторые временные ряды не содержат тенденции и циклической компоненты, а каждый следующий их уровень образуется как сумма среднего уровня ряда и некоторой (положительной или отрицательной) случайной компоненты.

В большинстве случаев фактический уровень временного ряда можно представить как сумму или произведение трендовой, циклической и случайной компонент. Модель, в которой временной ряд представлен как сумма перечисленных компонент, называется аддитивной моделью временного ряда. Модель, в которой временной ряд представлен как произведение перечисленных компонент, называется мультипликативной моделью временного ряда. Основная задача статистического исследования отдельного временного ряда - выявление и придание количественного выражения каждой из перечисленных выше компонент с тем чтобы использовать полученную информацию для прогнозирования будущих значений ряда.

Для прогнозирования временных рядов чаще всего используют методы скользящей средней, экспоненциального сглаживания, Холта, Холта-Винтера, *ARIMA* (*АРПСС* – авторегрессия проинтегрированного скользящего среднего) [5]. Все они, в том числе и нейронные сети, реализованы в системе *Statistica* и многих других программных продуктах. Выбор метода зависит, в первую очередь, от поставленной цели, характера и параметров временного ряда.

В данных методических указаниях в качестве альтернативы предлагаются динамические адаптивные модели для внутрифирменного планирования, реализованные в системе Matlab/Simulink. Теоретические основы методов прогнозирования временных рядов не рассматриваются, поскольку они имеются в справочниках и учебниках по статистике, а также широко представлены в Интернете.

Хотя все перечисленные модели относятся к адаптивным в том смысле, что учитывают текущую информацию, предлагаемые модели позволяют получить решение при оптимальных значениях весовых коэффициентов, соответствующих наилучшему качеству аппроксимации. Под аппроксимацией понимается сглаживание фактического временного ряда расчетной траекторией, предназначенной для прогнозирования следующего значения показателя. Наиболее распространенным критерием выбора оптимального варианта в теории и практике управления является минимум среднего квадрата ошибки, то есть из каждого фактического значения вычитают соответствующие расчетные величины, возводят разницу в квадрат и находят среднее значение. При этом получаются довольно большие или, наоборот, очень малые числа, поэтому из полученного результата извлекают квадратный корень и получают абсолютную среднеквадратическую ошибку *RMS* (*roots-mean-square*). Относительную среднеквадратическую ошибку получают на основе отношения абсолютной текущей ошибки к фактическому значению. Кроме этого критерия используют и другие, полученные в виде средних абсолютных или относительных погрешностей прогноза. В данной работе для выбора варианта решения используется критерий *RMS*.

2 Математические модели динамических процессов и систем

2.1 Модели динамических систем

В современной математике используется представление динамических процессов и систем дифференциальными уравнениями в пространстве состояний. Такое описание процессов и систем позволяет легко проводить их цифровое моделирование, используя конечно-разностное представление и проектировать универсальные алгоритмы обработки информации с целью дальнейшего оптимального оценивания параметров систем и процессов. Оптимальные оценки необходимы для организации управления в системах автоматического управления современными методами, а в информационно-измерительных системах для получения достоверных данных об измеряемых физических величинах, для прогнозирования поведения исследуемых явлений и систем, повышения отказоустойчивости обработки информации. Одним из методов получения математической модели системы или процесса является идентификация.

Идентификацией динамической системы называется получение или уточнение по экспериментальным данным математической модели (числовых параметров) этой системы или процесса, выраженной посредством того или иного математического аппарата.

Используются следующие основные математические модели в пространстве состояний [2,6].

Непрерывная детерминированно-стохастическая динамическая система (ДС) – это система, описываемая линейными дифференциальными уравнениями состояния первого порядка и линейным уравнение выхода. В матричном виде

$$X'(t) = A \cdot X(t) + B \cdot U(t) + D \cdot V(t), \quad Y(t) = C \cdot X(t),$$

где $X'(t)$ - n-мерный вектор состояния системы; $V(t)$ - r-мерный вектор гауссовских шумов с нулевым средним и корреляционной матрицей $E[V(t)V^m(t)]=Q(t)$, (E – оператор математического ожидания); $Y(t)$ – m-мерный вектор выхода; A, B, D – матрицы состояния (матрицы коэффициентов); C – матрица линейного преобразования размера m x n.

Дискретная детерминированно-стохастическая динамическая система (ДС) – это система, описываемая разностными уравнениями первого порядка состояния и дискретным уравнением выхода. Матричный вид соответствует уравнениям

$$X(k+1)=F\cdot X(k)+G\cdot U(k)+T\cdot V(k),\ \ Y(k)=C\cdot X(k),$$

где $F, G, T,$ – переходные матрицы. Матрицы F, G, T вычисляются через A, B, D в виде $F=I+A\cdot\psi\cdot dt,\ G=\psi\cdot B\cdot dt,\ T=\psi\cdot D\cdot dt$,

где $\psi = I + A\cdot\dfrac{dt}{2} + A^2\cdot\dfrac{dt^2}{6} + A^k\cdot\dfrac{dt^k}{(k+1)!} + \ldots,$

где I – единичная матрица;

dt – период дискретности системы (процесса). Период дискретности dt выбирается исходя из полосы пропускания ДС в соответствии с импульсной теоремой.

Детерминированной является ДС, у которой отсутствуют шумы возмущения и нет стохастических процессов (или всеми этими факторами можно пренебречь). У чисто стохастической ДС отсутствует детерминированный вектор входных сигналов. Детерминировано-стохастическая система содержит как детерминированные воздействия, так и стохастические процессы.

Объектами наблюдения динамических систем являются: информационные процессы (ИП), объекты управления (ОУ), датчики первичной информации (ДПИ), исполнительные устройства (ИУ). Первичной моделью объекта наблюдения типа ИП является спектральная или корреляционная функция. Первичной моделью объекта наблюдения типа ОУ, ДПИ и ИУ является дифференциальное уравнение (или эквивалентная передаточная функция), связывающая вход и выход.

Датчик первичной информации – это элемент устройства, преобразующий информацию о физической величине в сигнал, удобный для использования и обработки. Он задается дифференциальным уравнением или передаточной функцией. Передаточной функцией ДПИ является отношение преобразования Лапласа выходного процесса ДПИ к преобразованию Лапласа входного процесса при нулевых начальных условиях.

Движением системы называется физический процесс изменения её переменных во времени и пространстве. Выходные переменные Y(t), управляющие входные воздействия U(t) и возмущающие входные воздействия V(t) рассматриваются в виде соответствующих векторов, которые записываются в виде столбцовых матриц

$$Y(t)=[y1, y2,..., ym]^T,\ U(t)=[u1, u2,..., us]^T,\ V(t)=[v1, v2,..., vr]^T.$$

2.2 Моделирование непрерывной системы контроля

Система контроля предназначена для измерения и выдачи информации о контролируемом процессе h(t), который содержит среднюю (детерминированную) составляющую и стохастическую (случайную) g(t). Измерение происходит при воздействии аддитивных шумов n(t). Датчик, с помощью которого производятся измерения, является динамическим звеном (в данном случае второго порядка). Эквивалентная схема системы контроля представлена на рисунке 2.1[4,7].

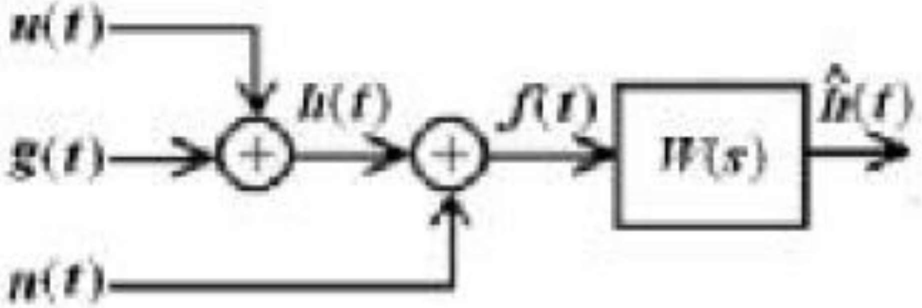


Рисунок 2.1 – Схема системы контроля

Случайная составляющая g(t) измеряемого процесса задана спектральной плотностью Sg(w); детерминированная – сигналом u(t); h(t)=g(t)+u(t) – полный информационный процесс; f(t)=h(t)+n(t) – измерение процесса h(t) с аддитивными шумами n(t) (задана спектральная плотность шума - Sn(w)); h^(t) – выходной сигнал ДПИ (датчик первичной информации); W(S) – передаточная функция ДПИ. Детерминированное входное воздействие задано суммой ступенчатой и гармонической функций.

2.3 Математическое описание непрерывной системы контроля

Задана спектральная плотность контролируемого процесса

$$S(w) = \frac{\omega^2 + 16}{\omega^4 + \omega^2 + 4}.$$

Передаточная функция объекта наблюдения

$$W(s) = \frac{0{,}45}{2 \cdot s^2 + 0{,}18 \cdot s + 0{,}28}.$$

Интенсивность шумов измерений R=17 (при измерении выходного сигнала объекта наблюдения).

Путем факторизации из модели в виде спектральной плотности получим передаточную функцию формирующего фильтра входного процессора

$$.W_g(s) = \frac{4 + s}{s^2 + 2{,}236 \cdot s + 2}.$$

Матричная модель объекта наблюдения находится методом вспомогательной переменной. Уравнение состояния в данном случае

$$.\ X' = A \cdot X + B \cdot u + D \cdot \upsilon.$$

Процесс h(t) на выходе объекта наблюдения вычисляется в матричном виде $g = C_g \cdot X$.

В данном примере получаем следующий вид матриц

$$A_g = \begin{bmatrix} 0 & 1 \\ -2 & -2{,}236 \end{bmatrix}; \quad B_g = \begin{bmatrix} 0 \\ 1 \end{bmatrix}; \quad C_g = \begin{bmatrix} 4 & 1 \end{bmatrix}.$$

Матричная модель датчика $\dot{X}_s = A_s \cdot X_s + B_s \cdot g$ имеет вид

$$A_0 = \begin{bmatrix} 0 & 1 \\ -0{,}14 & -0{,}09 \end{bmatrix}; \quad B_0 = \begin{bmatrix} 0 \\ 0{,}5 \end{bmatrix}; \quad C_0 = \begin{bmatrix} 0{,}25 & 0 \end{bmatrix}.$$

Выход объекта наблюдения $h = C_0 \cdot X_0$.

Полное уравнение объекта контроля содержит уравнение состояния входного процесса и уравнение состояния объекта

$$X' = A \cdot X + B \cdot u + D \cdot \upsilon,$$

где матрицы *A*, *B* и *D* составляются на основе дифференциальных уравнений процесса и объекта контроля, которые имеют вид

$$\begin{cases} \dot{X}_{1g} = X_{2g}, \\ \dot{X}_{2g} = -2X_{1g} - 2{,}236X_{2g} + \upsilon, \\ \dot{X}_{1o} = X_4 \\ \dot{X}_{2o} = -0{,}14X_{1o} - 0{,}09X_{2o} + 0{,}5(g + u), \end{cases}$$

. или относительно полного вектора $X = \{X_1; X_2; X_3; X_4\} = \{X_{1g}; X_{2g}; X_{1o}; X_{2o}\}$

$$\begin{cases} \dot{X}_1 = X_2, \\ \dot{X}_2 = -2X_1 - 2{,}236X_2 + \upsilon, \\ \dot{X}_3 = X_4 \\ \dot{X}_4 = -0{,}14X_3 - 0{,}09X_4 + 2X_1 + 0{,}5X_2 + 0{,}5u. \end{cases}$$

Матрицы A, B, C, D в данном случае имеют следующий вид

$$A=\begin{bmatrix} 0 & 1 & 0 & 0 \\ -2 & -2.236 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 2 & 0.5 & -0.14 & -0.09 \end{bmatrix}; B=\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0.5 \end{bmatrix}; C=\begin{bmatrix} 0 & 0 & 0.255 & 0 \end{bmatrix}; D=\begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}.$$

2.4 Математическое описание модели Холта

Данный метод относится к типу адаптивных методов и применяется для прогнозирования по временным рядам с явно выраженными трендами.

Сглаживание исходного ряда осуществляется с помощью двух экспоненциальных фильтров:

– фильтра данных с параметром $0 \le \alpha \le 1$;

– фильтра тренда с параметром $0 \le \beta \le 1$.

Для операции сглаживания и прогнозирования используется рекуррентная модель $Z_{i+1} = A_i + B_i$, $i = \overline{2, n}$,

где $A_i = \alpha Y_i + (1 - \alpha)(A_{i-1} + B_{i-1})$,

$B_i = \beta(A_i - A_{i-1}) + (1-\beta)B_{i-1})$,

Y_i – исходный временной ряд,

Z_i – сглаженный ряд.

Проблема начальных условий. На практике принимается

$A_2=Y_1$ и $B_2=Y_2$-Y_1.

Параметры фильтров задаются, или вычисляются методом наименьших квадратов $S(\alpha,\beta) = \sum_{i=1}^{n-2}[Y_{i+2} - Z_i(\alpha,\beta)]^2 \to \min$.

Модель Холта отличается от простого экспоненциального сглаживания тем, что учитывает так называемый локальный тренд, то есть изменение цепных абсолютных приростов во времени [5,8].   С этой целью кроме параметра альфа вводится новый параметр экспоненциального сглаживания бета и соответственно единица минус бета. Тогда сглаженную функцию $c_0(t)$ рассчитывают по формуле

$$c_0(t) = \alpha x_t + [(1-\alpha)\{c_0(t-1) + c_1(t-1)\}],$$

где x_t – фактический уровень ряда в момент t.

Функция $c_1(t)$, учитывающая локальные тренды

$$c_1(t) = \beta[c_0(t) - c_0(t-1)] + (1-\beta)c_1(t-1).$$

Прогноз процесса X_t на q единиц вперед для модели Холта вычисляется по формуле $X_{t+k} = c_0(t) + c_1(t)q$.

3 Моделирование процессов и систем в пакете Simulink

3.1 Моделирование непрерывной системы контроля в пакете Simulink

Для моделирования системы контроля в Matlab составляется схема моделирования, которая представлена на рисунке 3.1[9].

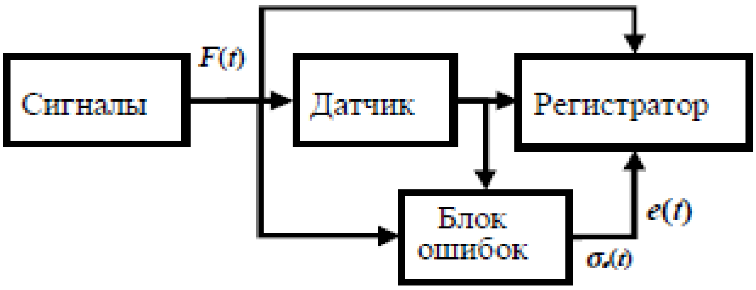


Рисунок 3.1 – Схема моделирования системы контроля

Алгоритм выполнения работы в среде Simulink.

1. Запускаем Matlab (версия R2012b) и выбираем в меню пункт «New → Simulink Model» (рисунок 3.2).

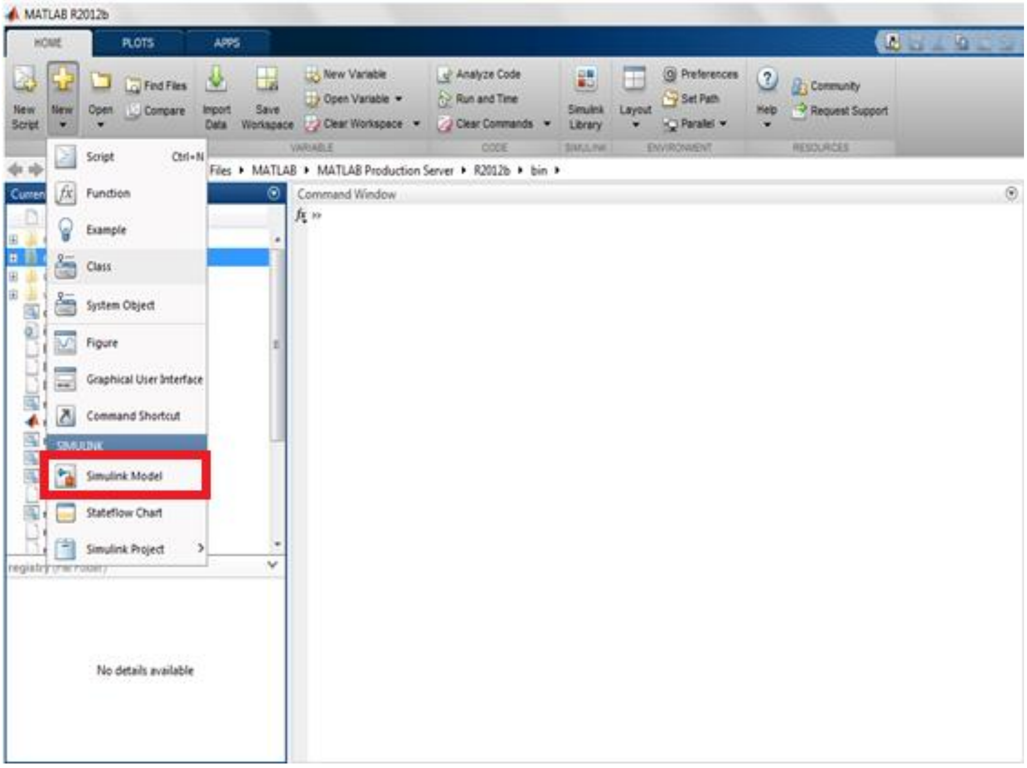


Рисунок 3.2 – Процесс создания новой модели в Simulink

2. Открываем библиотеку функциональных блоков "Simulink". Для этого кликнем левой кнопки мыши на панели управления по иконке "Simulink Library" (рисунок 3.3).

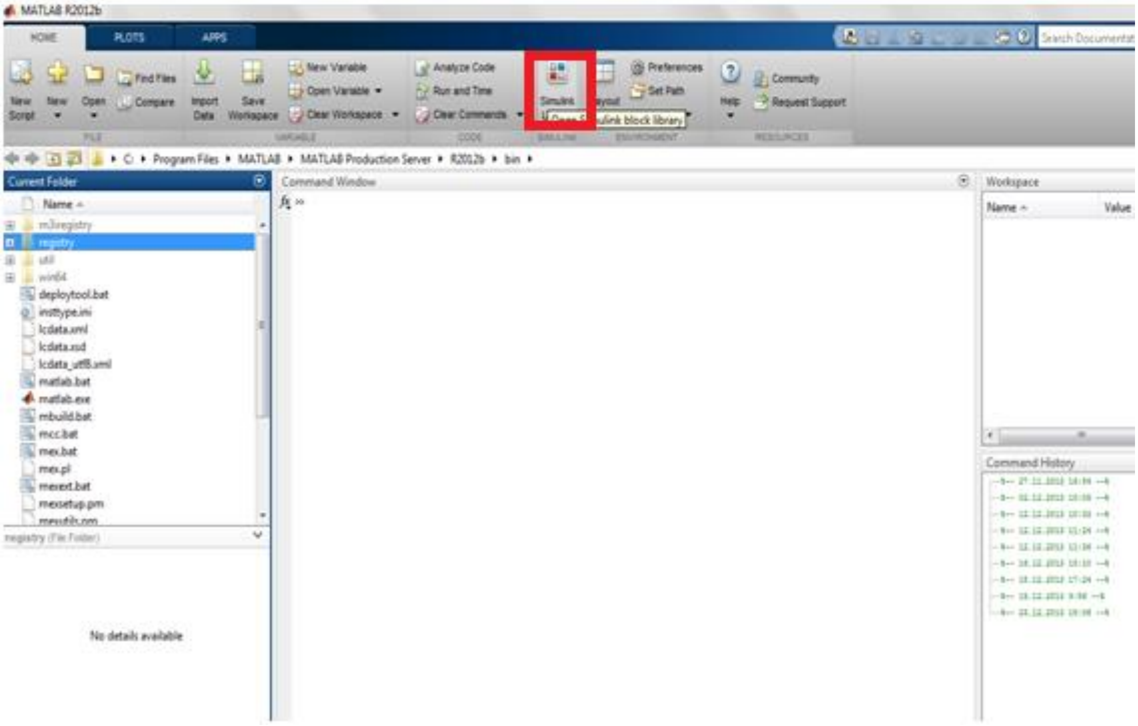


Рисунок 3.3 – Процесс создания новой модели в Simulink

3. В результате откроется меню библиотеки Simulink, главный вид которой представлен на рисунке 3.4.

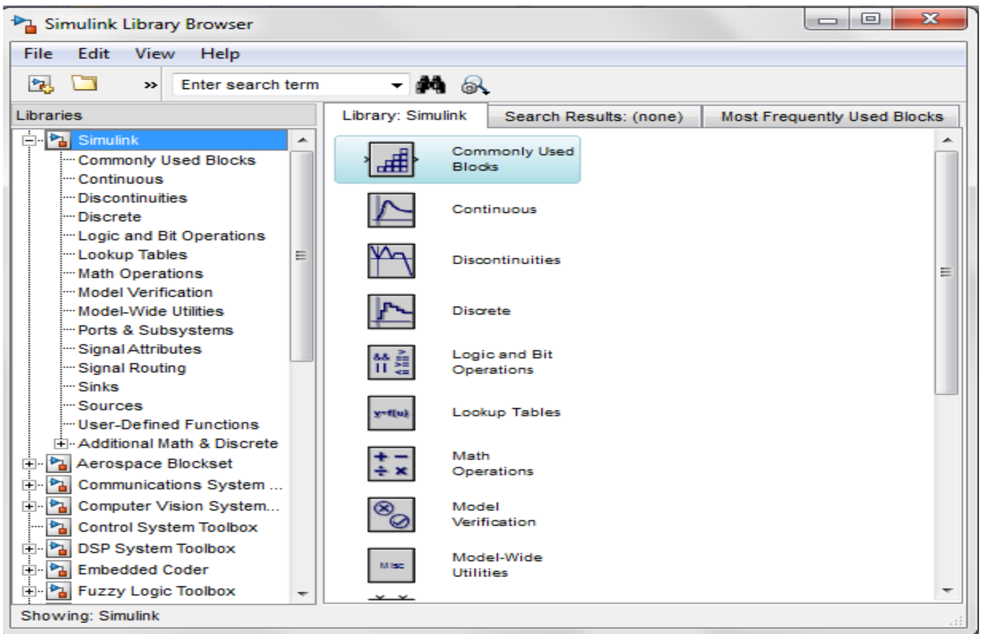


Рисунок 3.4 – Главное окно "Simulink Library"

4. Извлекаем из библиотеки Simulink все необходимые функциональные блоки. Для этого воспользуемся поиском, в верхней панели окна "Simulink Lybrary Browser", который представлен на рисунке 3.5.

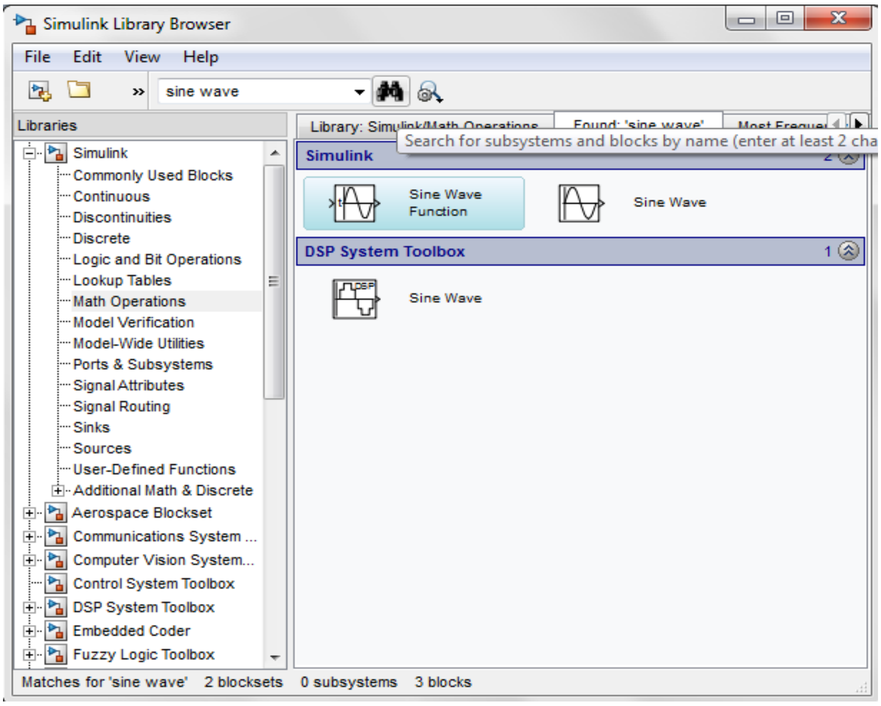


Рисунок 3.5 – Поиск блока в "Simulink Library"

5. Для моделирования непрерывной системы контроля нам будут необходимы следующие блоки:

– блоки "Sine Wave", "Step" и "Random number" с вкладки "Sources";

– три блока "Subsystem" и блок "Scope" с вкладки "Commonly Used Blocks";

– блок "Sum" с вкладки "Math Operations";

– блок "Fcn" с вкладки "User Define Function";

– блок "State-space" с вкладки "Continuous".

6. Соберем схему верхнего уровня модели непрерывной системы контроля (рисунок 3.6), используя перечисленные в п.5 функциональные блоки:

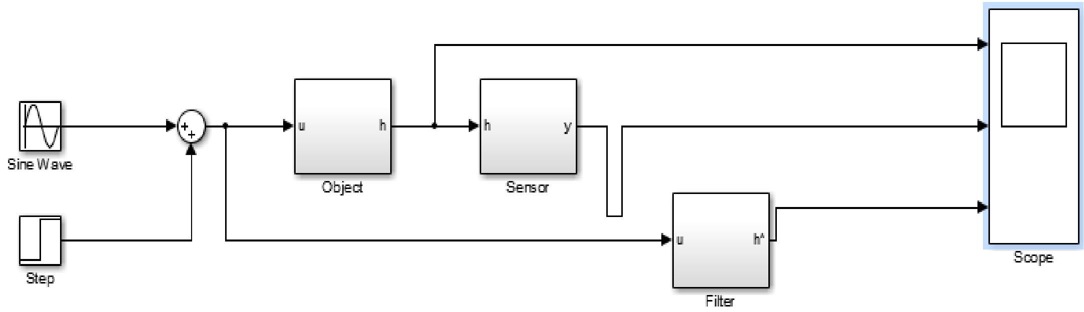


Рисунок 3.6 – Схема верхнего уровня системы контроля

7. Рассмотрим более подробно блоки "Subsystem": "Object", "Sensor", "Filter".

8. Блок "Object" является объектом наблюдения системы и представляет собой динамическую систему, в которой содержится стохастический процесс (блок "State-Space") и датчик (блок "State-Space 1"). Функциональная схема динамической системы "Object" представлена на рисунке 3.7.

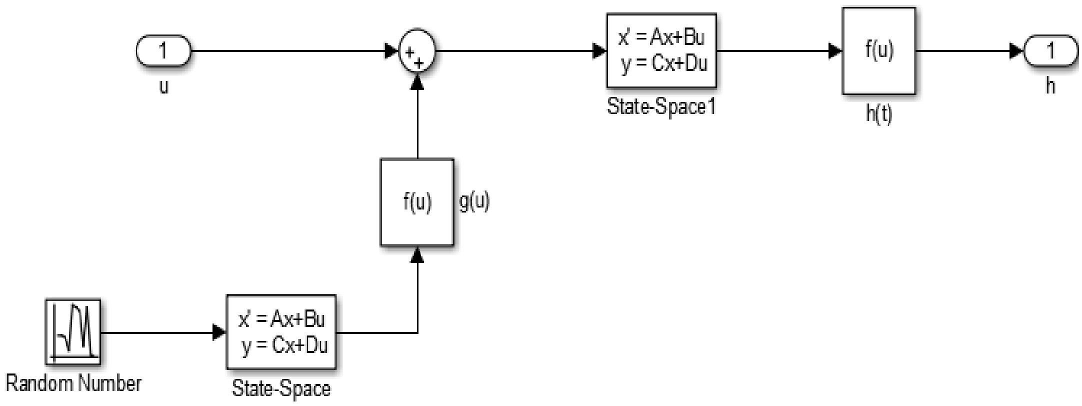


Рисунок 3.7 – Динамическая система "Object"

9. Настройка блоков уравнения состояния "State-Space" и "State-Space 1" представлена на рисунках 3.8 и 3.9 соответственно.

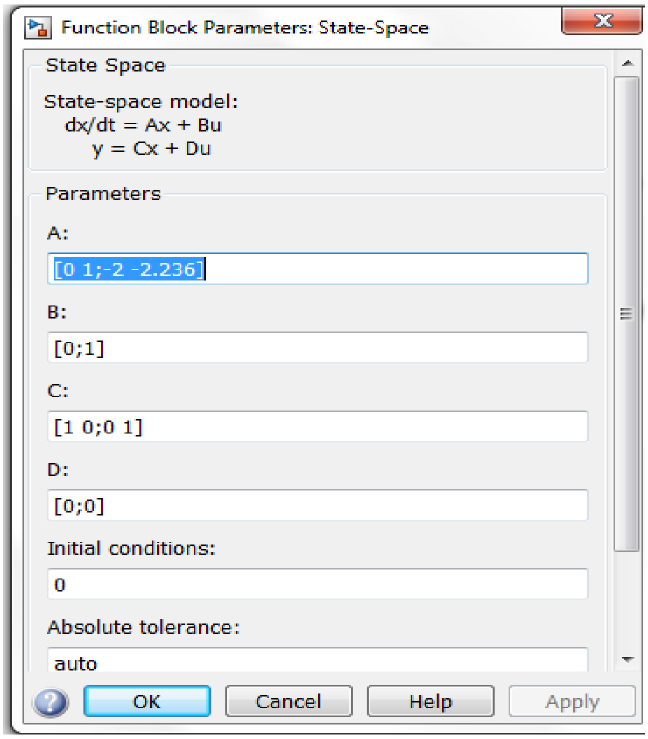


Рисунок 3.8 – Настройка параметров блока "State-Space"

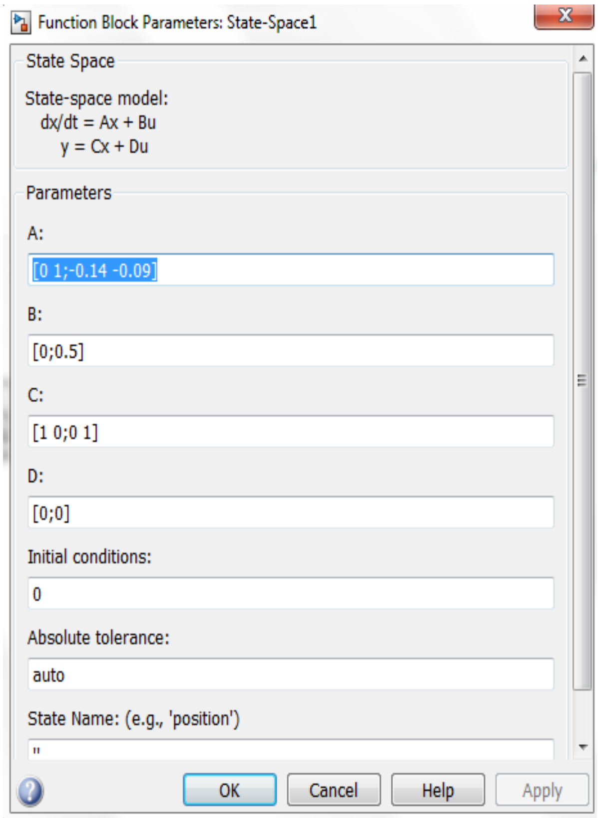


Рисунок 3.9 – Настройка параметров блока "State-Space 1"

10. Функциональные блоки $h(t)=C_0X$ и $g(u)=C_gX$, заданы функциями, представленными в окне параметров (рисунок 3.10).

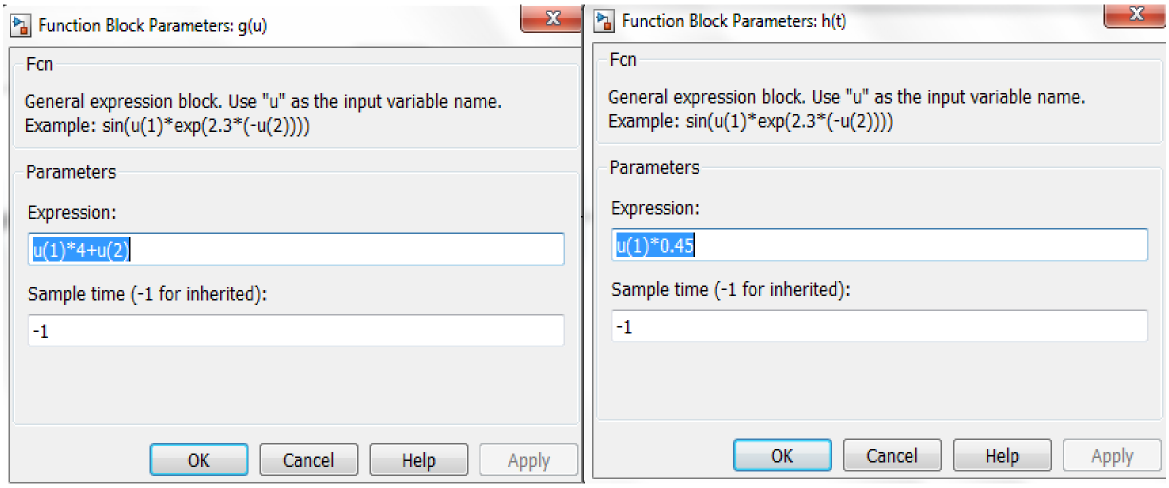


Рисунок 3.10 – Настройка функциональных блоков h(t) и g(u)

11. Блок "Sensor" (датчик) производит измерение входного сигнала и представляет собой совокупность полезного сигнала h(t) и помехи n(t):

$y(t)=h(t)+n(t)$.

Модель датчика представлена на рисунке 3.11[4,10]. Блок "Random Number" используется в качестве генератора белого шума с интенсивностью 0,4.

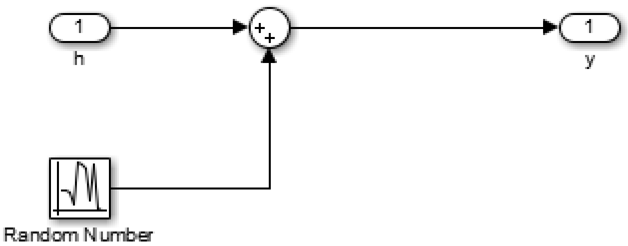


Рисунок 3.11 – Модель датчика (Sensor)

12. Блок "Filter" (фильтр) на основе измерений датчика выдает оценку выходного параметра объекта наблюдения - h^(t). Матрицы A, B, C соответствуют матрицам полной модели. Матрица C в блоке "State Space" - единичная. Модель фильтра представлена на рисунке 3.12.

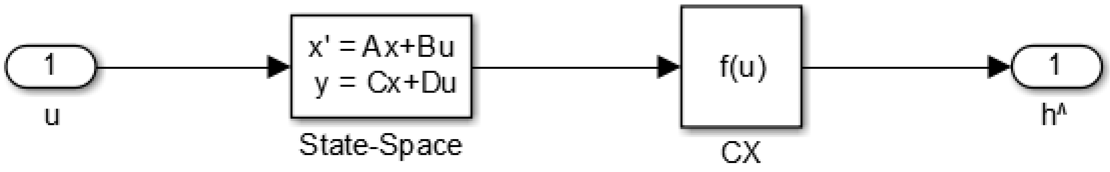


Рисунок 3.12 – Модель фильтра (Filter)

Настройка параметров блока "State Space" и функционального блока f(u) представлена на рисунке 3.13.

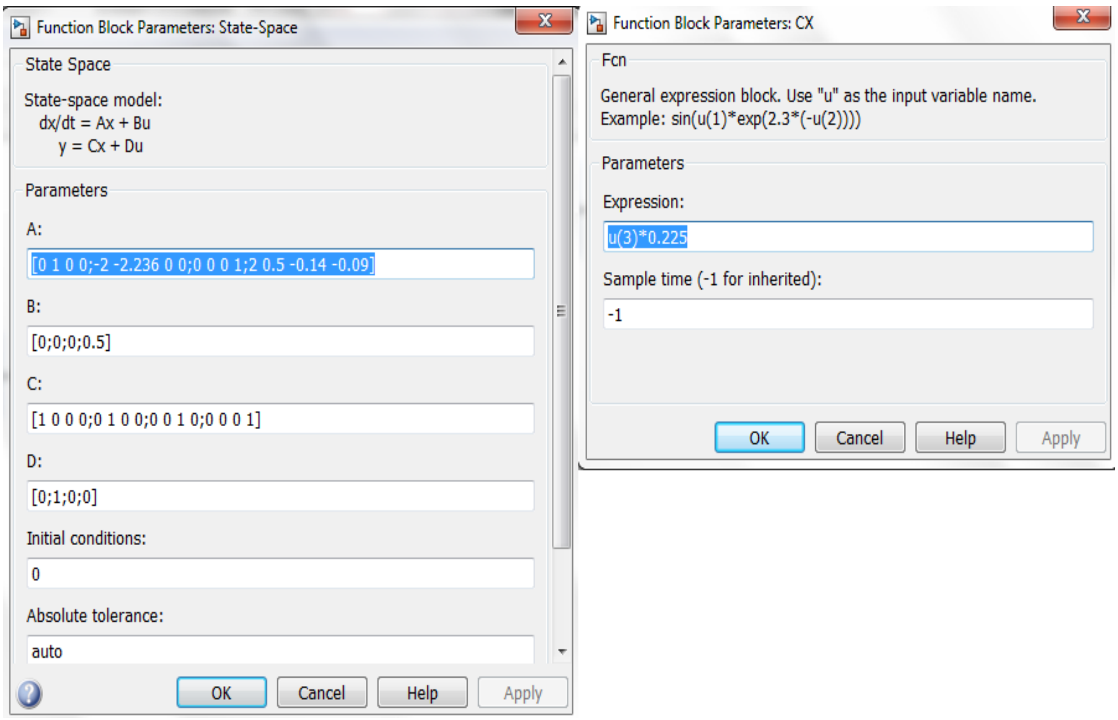


Рисунок 3.13 – Настройка параметров блоков "State-Space" и "f(u)"

13. Результаты процессов системы регистрируются осциллографом (блок "Scope"). Произведем настройку параметров блока "Scope". Для этого кликнем правой кнопкой мыши по блоку и выберем в диалоговом окне пункт "Block Parametres" (параметры блока). Далее в области появившегося окна кликнем правой кнопкой мыши и выберем пункт "Axes properties" (рисунок 3.14). В появившемся диалоговом окне зададим область значений (Y) для каждого из трех графиков (рисунок 3.15).

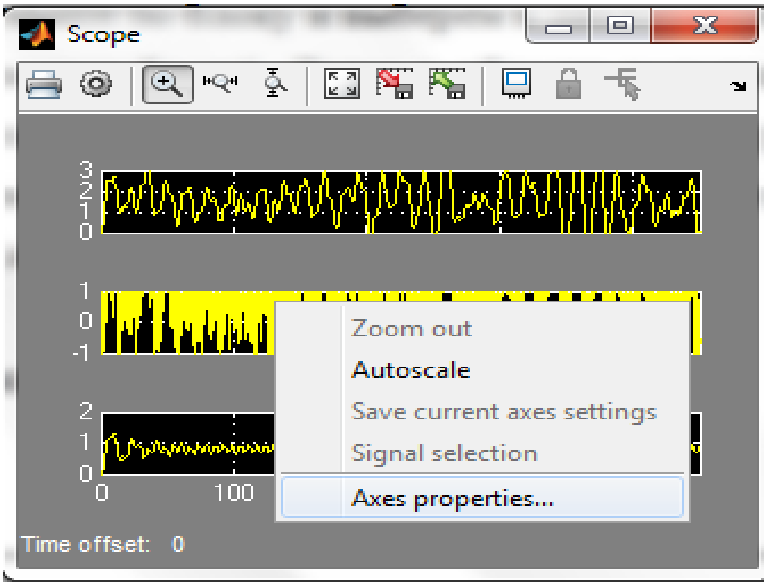


Рисунок 3.14–Настройка параметров блока "Scope"

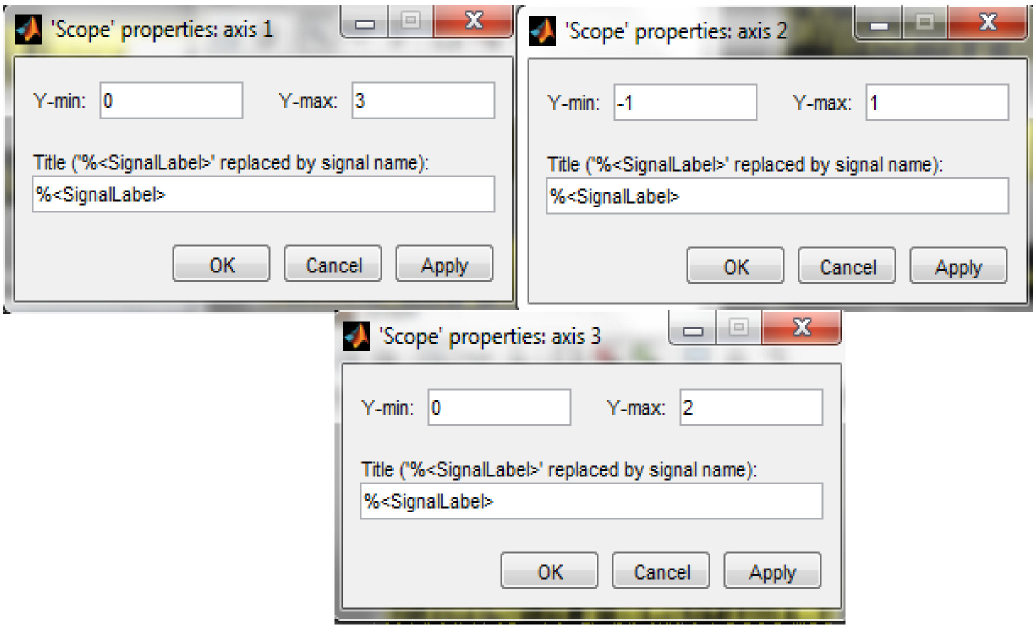


Рисунок 3.15 – Настройка области значений Y

14. На панели инструментов Matlab в верхней части экрана можно настроить число рабочих тактов системы, по окончании которых работа Matlab прекратится. Настройка данного параметра представлена на рисунке 3.16.

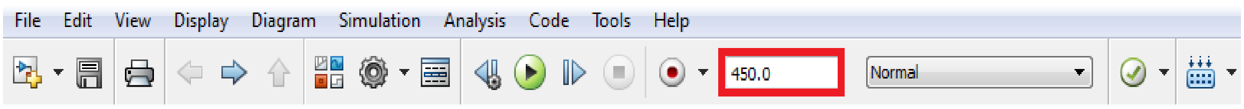


Рисунок 3.16 – Настройка рабочих тактов системы

15. На этом настройка модели непрерывной системы контроля завершена. Далее запустим систему, кликнув левой кнопкой мыши по иконке "Run" на панели инструментов в верхней части экрана (рисунок 3.17).

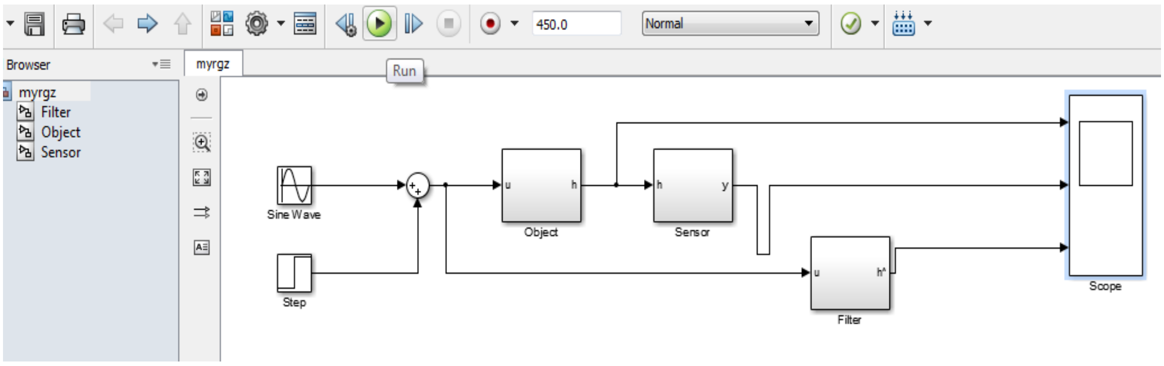


Рисунок 3.17 – Запуск системы на выполнение

16. Результаты работы системы отражаются в блоке "Scope" и приведены на рисунке 3.18.

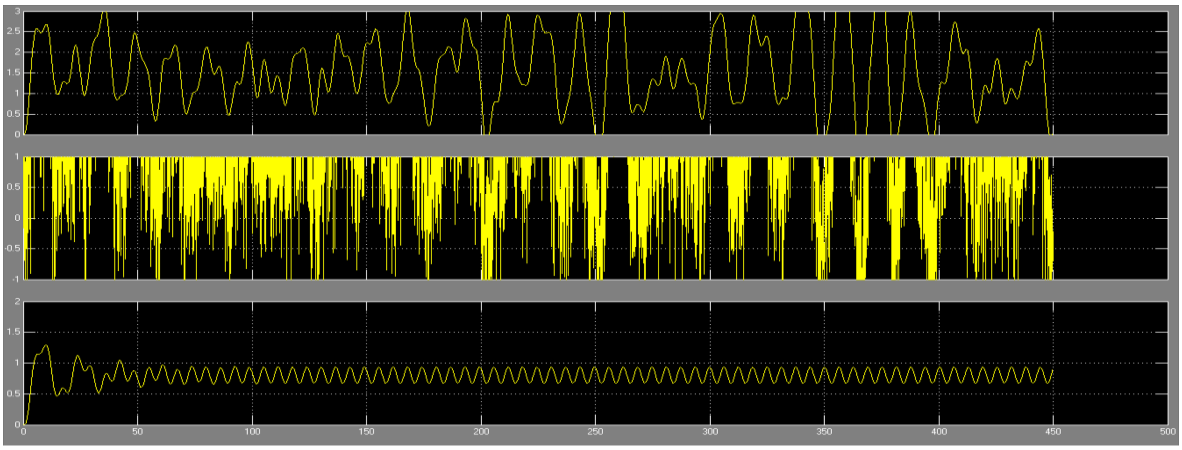


Рисунок 3.18 – Результаты работы системы

3.2 Моделирование динамической системы в среде Simulink

Используя дифференциальное уравнение $\frac{d^2y}{dx^2} + 2n(1+\alpha y^3)\frac{dy}{dx} + k^2(y+\beta y^2) = h\sin px,$, поставить и решить задачу Коши .

Используя данное дифференциальное уравнение, необходимо:

1) создать модель механической системы;

2) вычислить числовое значение координаты осциллятора в момент времени t=5 и вывести результат на display;

3) построить графики зависимости координаты и скорости от времени;

4) построить фазовую траекторию системы.

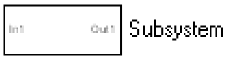
Алгоритм решения задачи.

1. Запишем исходное уравнение в виде системы уравнений первого порядка:

$$\begin{cases} \dot{y} = z, \\ \dot{z} = h \cdot \sin\, pt - 2 \cdot n \cdot (1 + \alpha \cdot y^3) \cdot z - k^2 (y + \beta^2). \end{cases}$$

Решим эту систему с помощью пакета Simulink, составляя блочную модель.

2. Запускаем Matlab и создать файл модели с помощью команды File → New → Model, или, используя кнопку  на панели инструментов.

3. Отдельным блоком в общей модели сформируем подмодель (блок Subsystem) 

(библиотека Ports & Subsystems).

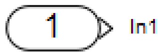
Подмодель – это фрагмент модели, оформленный в виде отдельного блока. Использование подмодели при составлении модели имеет следующие положительные стороны:

1) уменьшает количество одновременно отображаемых блоков на экране, что облегчает восприятие модели;

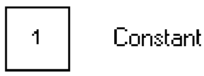
2) позволяет создавать и отлаживать фрагменты модели по отдельности, что повышает технологичность создания модели;

3) дает возможность синхронизации параллельно работающих подсистем.

Подмодель исходной задачи содержит функцию правой части первого уравнения системы: $f(t, y, \dot{y}) = h \cdot \sin\, pt - 2 \cdot n \cdot (1 + \alpha \cdot y^3) \cdot \dot{y} - k^2 (y + \beta^2)$. В качестве входных параметров используем значения $y(t)$ и $\dot{y}(t)$, а в качестве входных величин (констант) – h, p, n, k.

4. В окне обозревателя разделов библиотеки Simulink находим раздел: Commonly Used Blocks, где выбираем блок 

, левой клавишей мыши перетаскиваем его в созданное окно. В данной модели таких блоков 8: y, a, b, n, y`, h, p, k.

5. Далее из этого же раздела выбираем блок 

. Требуется изменить условия использования блока, для этого меняем его параметры, установленные программой “по умолчанию”. Необходимо дважды щелкнуть левой клавишей мыши, указав курсором на изображение блока. Откроется окно редактирования параметров данного блока (рисунок 3.19) (каждому блоку соответствует свое окно параметров).

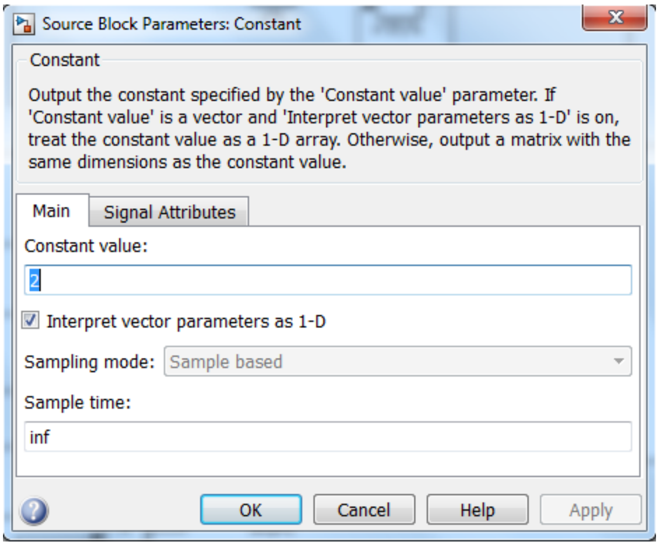
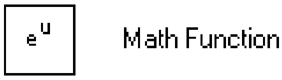
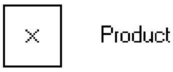


Рисунок 3.19– Параметры блока Constant

6. При создании модели для возведения $\boldsymbol{k}$ в квадрат используем блок вычисления математических функций «Math Function» 

. В окне «Параметры» блока выбираем параметр «Function». Из появившегося списка выбираем вид вычисляемой функции «magnitude^2» – вычисление квадрата.

7. Для операции умножения $\boldsymbol{k^2 y}$ используем блок 

.

8. В результате получим подмодель нелинейного осциллятора, представленную на рисунке 3.20.

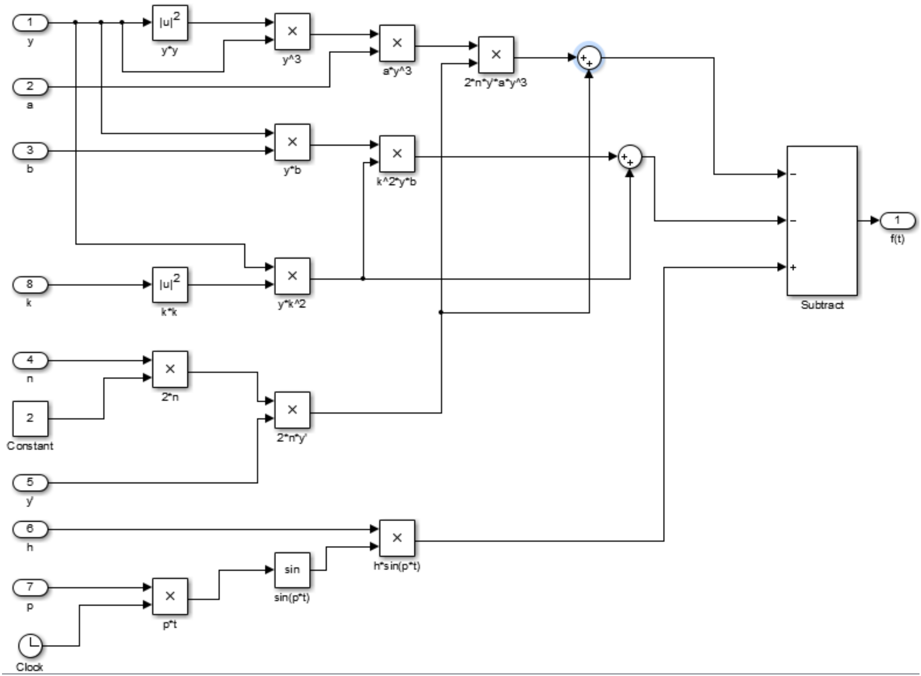


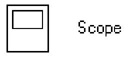
Рисунок 3.20 – Подмодель нелинейного осциллятора

9. Используя созданную подмодель, значения $y(t)$ и $\dot{y}(t)$ в основной модели связываем с соответствующими входами подмодели, а выход подмодели связываем

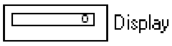
с сумматором. Сигнал с выхода сумматора подаем на вход первого интегратора, замыкая цепь интегрирования.

10. На выход первого интегратора подается ускорение $\dot{z}$, а в качестве начального условия используется начальное значение скорости $z(0)$. Выходом этого блока будет скорость осциллятора $z = z(t)$.

11. Величину скорости осциллятора подаем на вход второго интегратора с начальным условием в виде начального отклонения $\boldsymbol{y(0) = y_0}$. Выход из блока представляет собой закон движения $\boldsymbol{y = y(t)}$.

12. Для наблюдения за изменениями сигналов в процессе моделирования необходимо добавить блок осциллограф Scope 

, который строит графики исследуемых сигналов в функции времени.

13. Для отображения значения сигнала в виде числа используем блок 

.

В Simulink описанная процедура выглядит согласно рисунку 3.21.

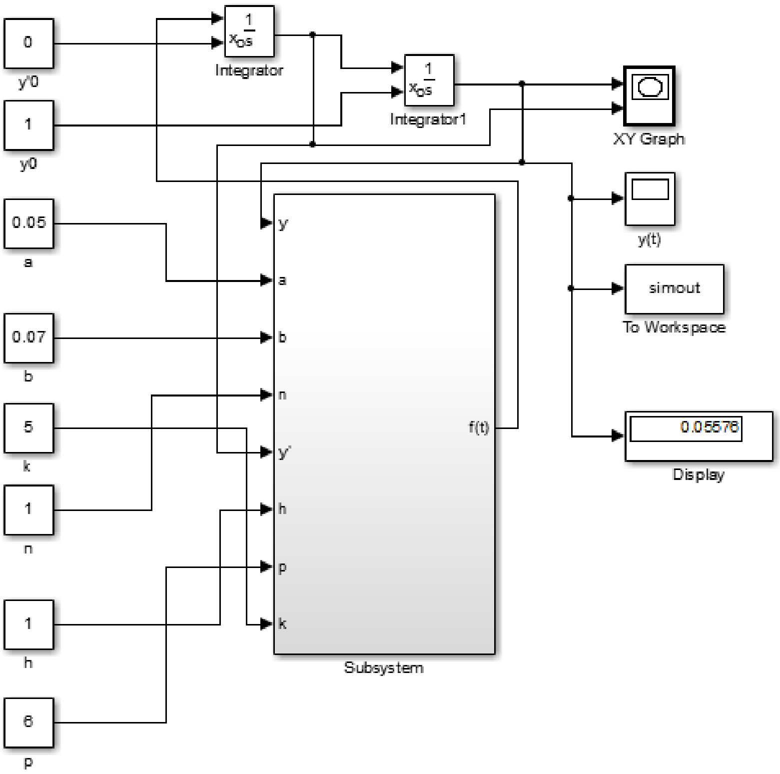


Рисунок 3.21– Основная модель нелинейного осциллятора

Если дважды щелкнуть мышью на блоке **Scope** ($y(t)$) в блок-схеме осциллятора, то появится графическое окно с графиком зависимости координаты y от времени. Результат показаний блока "Scope" представлен на рисунке 3.22.

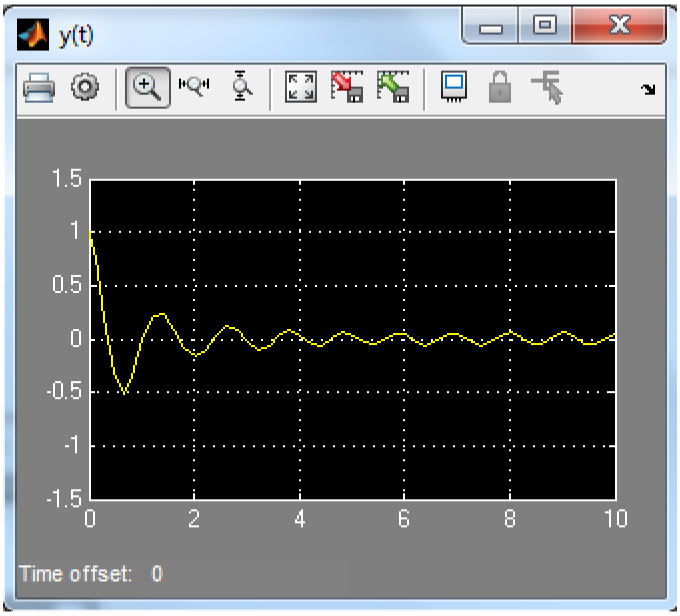
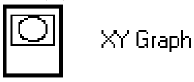


Рисунок 3.22 – Показания блока Scope колебания нелинейного осциллятора

14. В данной модели для построения фазовой траектории системы используется блок 

– графопостроитель, который строит график одного сигнала в функции другого (график вида Y(X)). Блок имеет два входа. Верхний вход предназначен для подачи сигнала, который является аргументом (X), нижний вход – для подачи значений функции (Y). Зависимость X от Y представлена на рисунке 3.23.

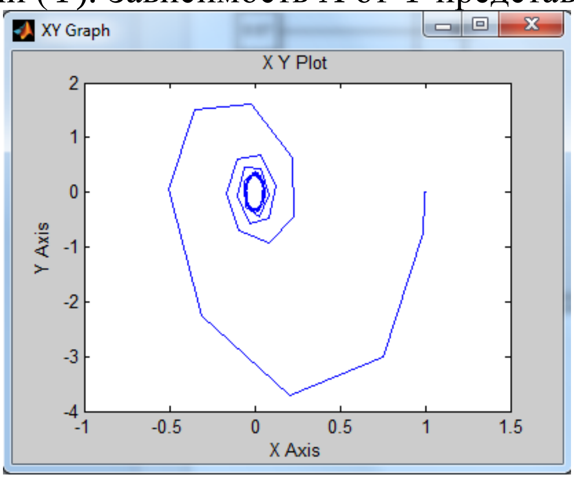


Рисунок 3.23 – Зависимость X от Y

3.3 Моделирование временных рядов методом Холта в среде Simulink

Для моделирования системы контроля в Matlab составляется блочная структура модели Холта, которая представлена на рисунке 3.24 [5,8].

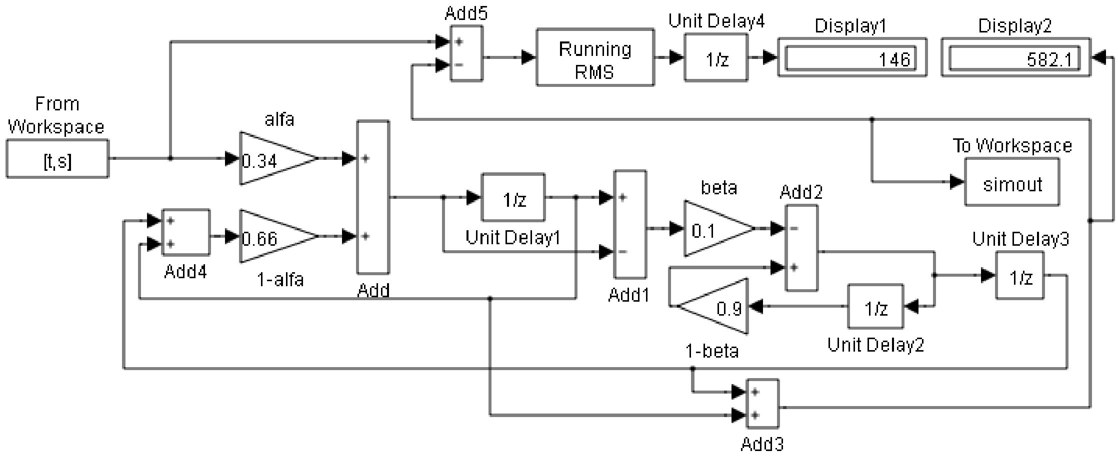


Рисунок 3.24 – Модель Холта

При использовании модели Холта также возникает проблема начальных условий. Существуют различные формальные процедуры их определения, которые вследствие недостаточного обоснования не дают существенного эффекта. Поэтому в данной модели начальный уровень расчетного ряда принят равным фактическому уровню, а начальный тренд – нулю.

Модель Холта приводит даже к большей ошибке по выбранному критерию, чем оптимальное однопараметрическое экспоненциальное сглаживание. Однако если за начальный уровень сглаженного ряда принять среднее значение за первые четыре квартала (375, блок *Uniti Delau2*), а за начальный тренд – средний абсолютный прирост за период, включая первый квартал следующего года (-12,5, блок *Uniti Delau3*), то ошибка уменьшается до 142,6 при почти таком же прогнозе. Кроме того результат прогнозирования в абсолютном выражении (дисплей 2) даже без изменения начальных условий ближе к фактическому значению. Для оценки начальных условий кроме приведенного способа можно найти уравнение регрессии для первых пяти-шести наблюдений, тогда отрезок (параметр смещения) принимается за начальный уровень, а наклон (угловой коэффициент) – за начальный тренд. Хотя эта операция в *Excel* занимает доли секунды, но существенных улучшений качества прогноза она тоже не дает. В системе *Statistica* начальный линейный тренд для модели Холта определяется как отношение разности конечного и начального фактических уровней ряда к числу наблюдений без единицы, а начальный уровень сглаженного ряда принимается равным фактическому начальному уровню минус половина расчетного тренда.

Алгоритм выполнения работы в среде Simulink.

1. Запускаем Matlab и выбираем в меню пункт «NEW → Simulink Model», как показано на рисунке 3.25.

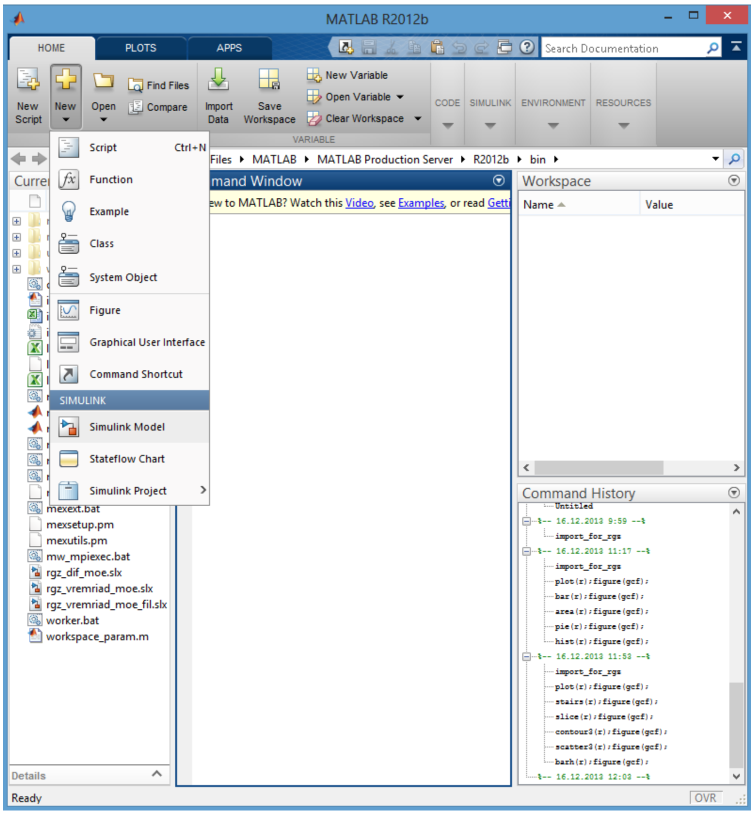


Рисунок 3.25 – Процесс создания новой модели в Simulink

2. Открываем библиотеку «Simulink → Open Simulink block library», согласно рисунку 3.26

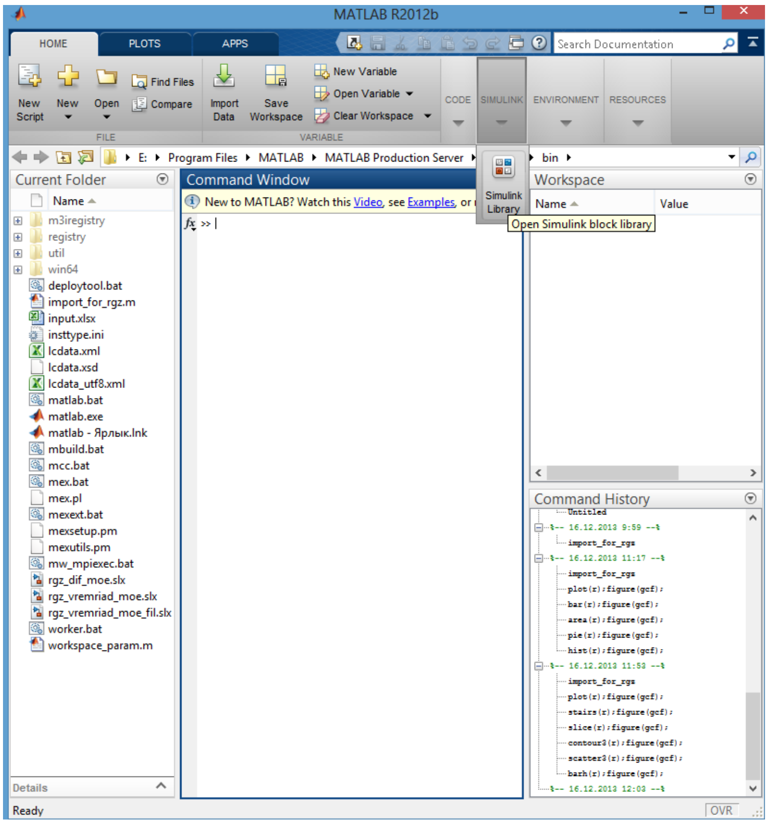


Рисунок 3.26 – Процесс создания новой модели в Simulink

3. Извлекаем из библиотеки (можно через поиск) все необходимые структурные элементы, как показано на рисунке 3.27.

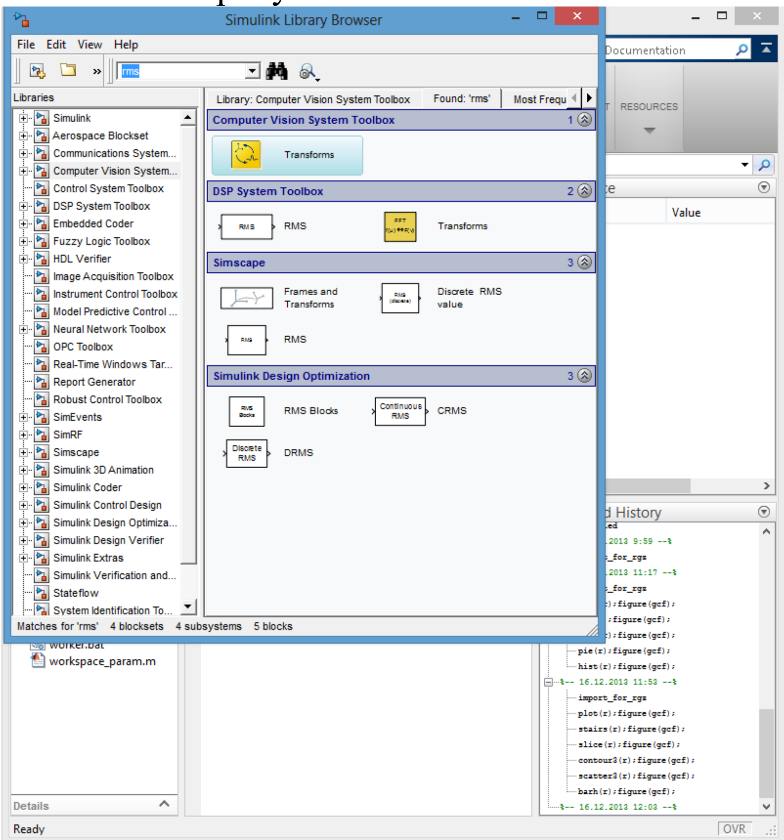


Рисунок 3.27 – Процесс создания новой модели в Simulink

4. Собираем выбранные элементы в соответствующую схему и сохраняем в рабочую папку Matlab под любым именем (сохранение можно произвести в любую папку компьютера, но из рабочей папки Matlab вам будет удобнее работать со схемой в дальнейшем). Пример схемы приведен на рисунке 3.27. Процесс сохранения созданной модели показан на рисунке 3.28.

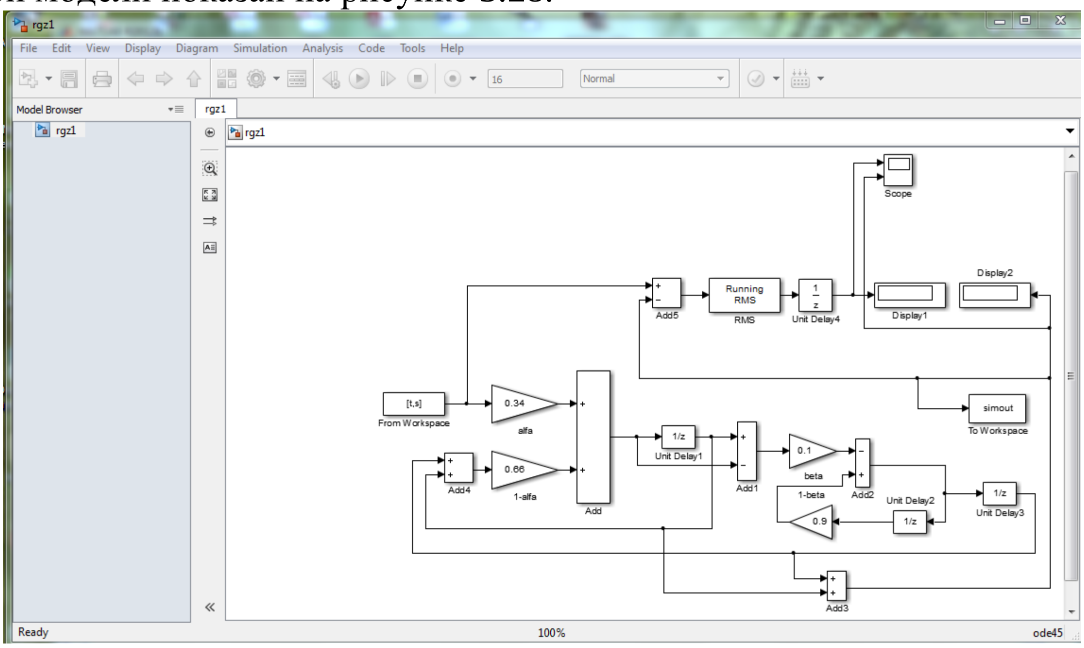


Рисунок 3.27– Пример построенной модели в Simulink

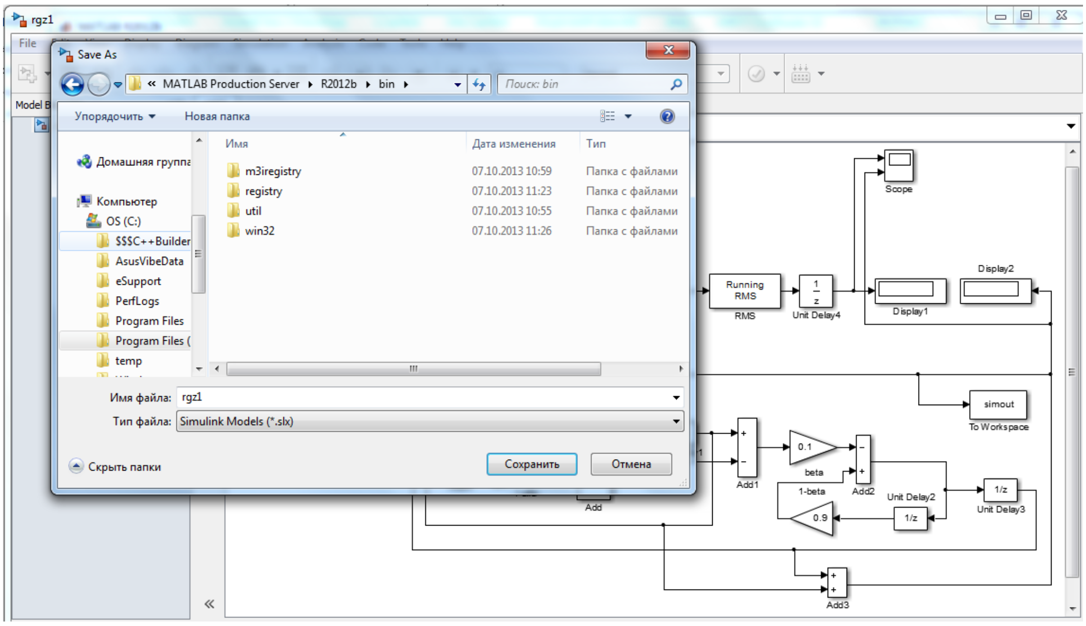


Рисунок 3.28 – Процесс сохранения новой модели в Simulink

5. Настройка блока исходных данных «FromWorkspace». Открываем меню настройки параметров (правый клик мыши и выбор пункта «Block Parameters» либо двойной щелчок левой кнопкой мыши). Процесс настройки параметров показан на рисунках 3.29 – 3.30.

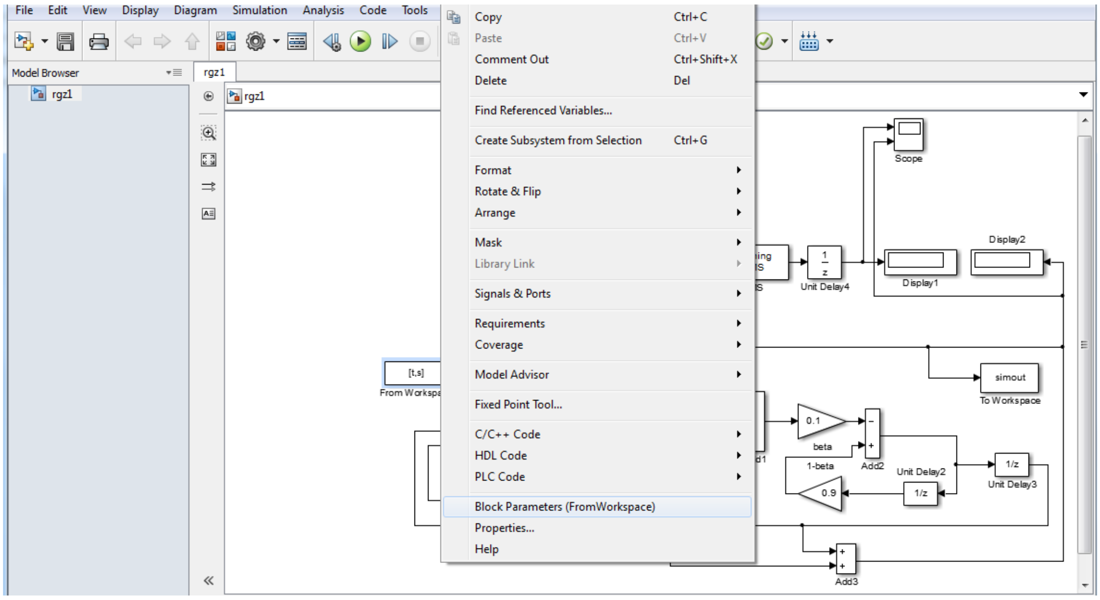


Рисунок 3.29 – Настройка блока исходных данных новой модели в Simulink

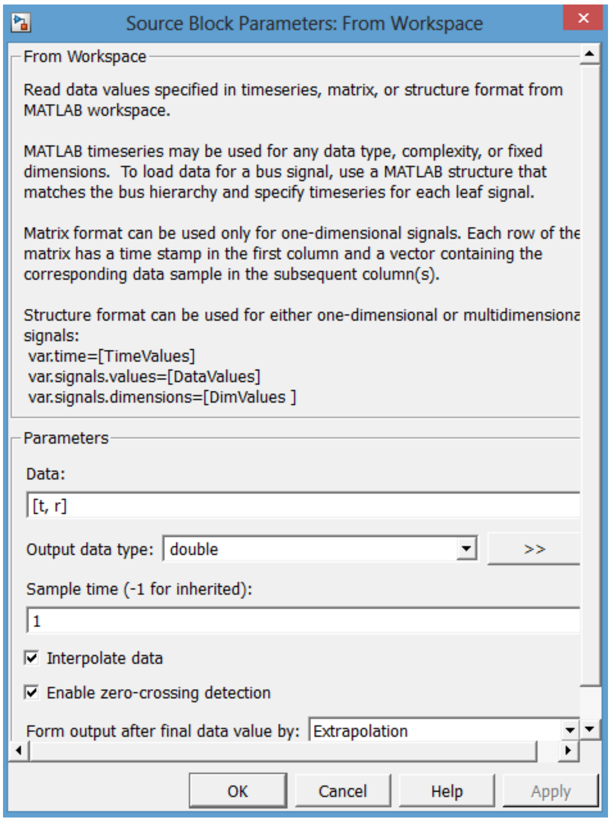


Рисунок 3.30 – Окно настройки блока исходных данных

В появившемся окне нас интересуют только два поля:

а) поле «Data». В этом поле нам необходимо указать имена входных

переменных в произвольном порядке (единственное исключение – переменная времени всегда должна стоять на первом месте), как показано на рисунке 3.31;

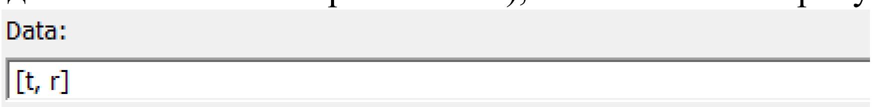


Рисунок 3.31 – Поле параметра «Data»

в) поле «Sample time». В этом поле мы указываем значение дискретного шага для временной последовательности. Рекомендуется установить его равным 1, как показано на рисунке 3.32.

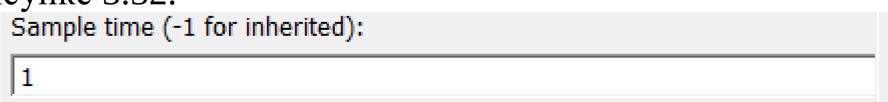


Рисунок 3.32 – Поле параметра «Sample time»

6. Задание и импорт входных данных в рабочую среду Matlab. Существует несколько способов для ввода данных в рабочую среду Matlab, нами в данной работе будет рассмотрен один из наиболее простых и, в тоже время, гибких способов задания данных:

1) создаем в рабочей папке Matlab лист MSExcel, согласно рисунку 3.33. В нем в виде колонок векторов данных задаем необходимые нам последовательности данных в произвольном порядке (вектор времени должен быть равномерно возрастающим и стоять на первом месте);

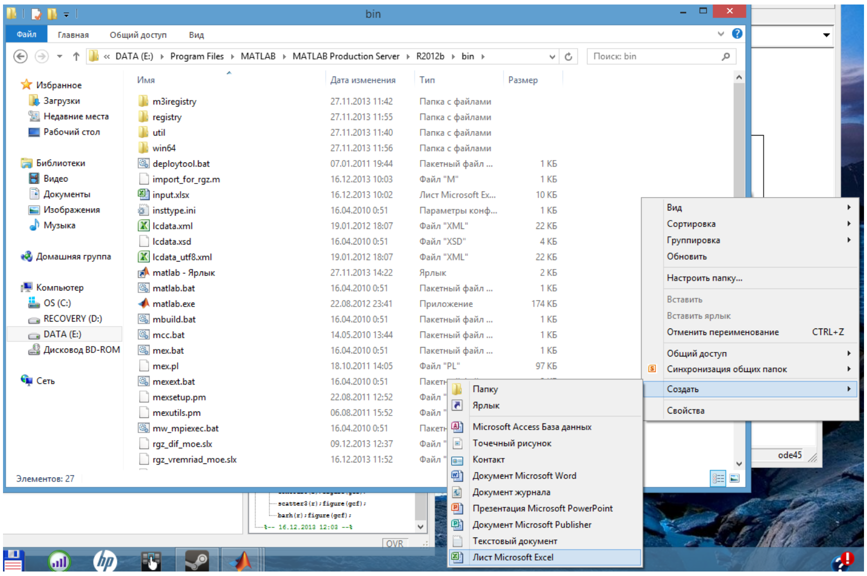


Рисунок 3.33 – Создание файла входных данных

2) сохраняем этот файл , согласно рисунку 3.34 под любым именем (например «input») в рабочую папку Matlab (опять же – файл можно сохранить в любой каталог компьютера, но для удобства дальнейшей работы его проще сохранить в рабочую папку Matlab);

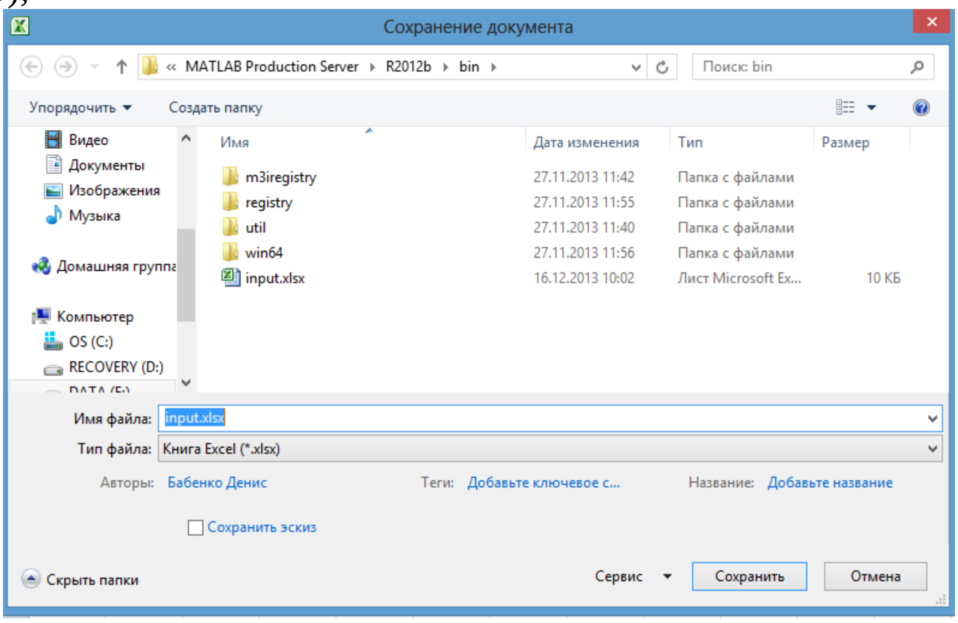


Рисунок 3.34 – Сохранение файла входных данных

3) теперь осуществим собственно импорт данных. В рабочей области Matlab выбираем «Import data from file» и выбираем наш файл MSExcel, с исходными данными. После этого Matlab сгенерирует таблицу, внешне напоминающую таблицу MSExcel. В этой таблице нам представлены колонки векторов данных (в данном случае t и r) с выделенной областью для импорта (по умолчанию Matlab выделяет все данные, а первые значения столбцов – считаются именами векторов, однако мы можем регулировать область импорта сами). Процесс импорта данных показан на рисунках 3.35 – 3.37.

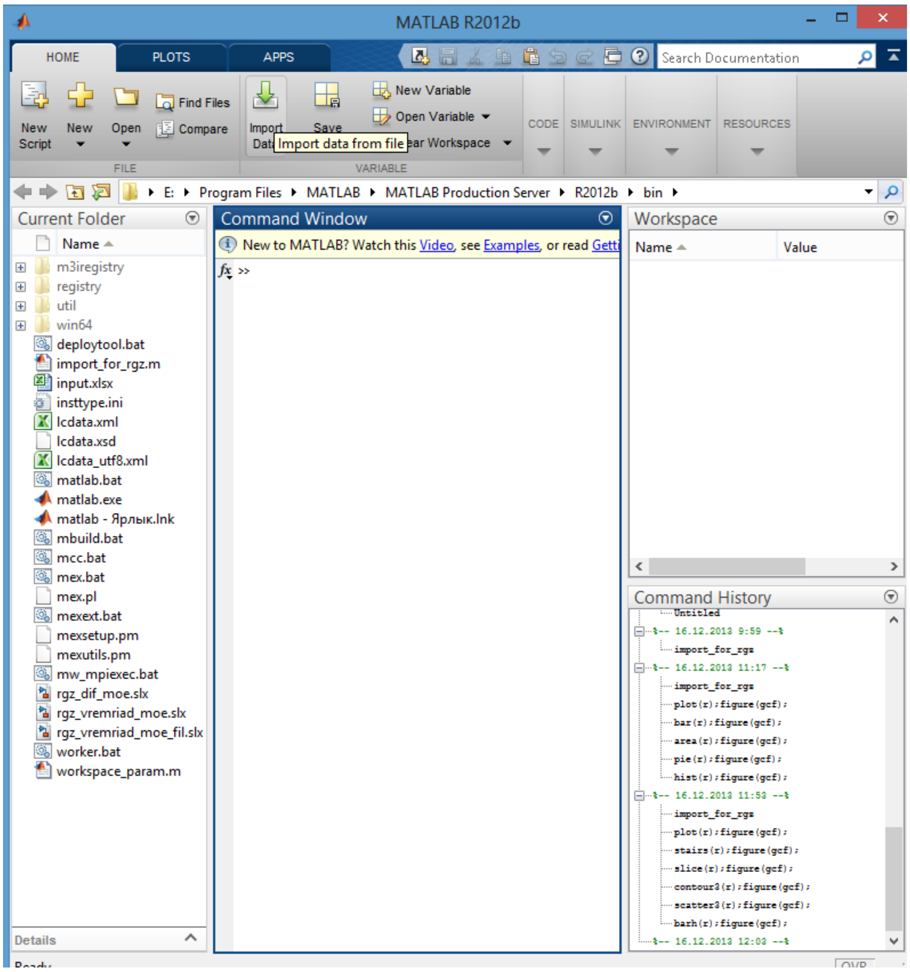


Рисунок 3.35 – Процесс создания модели в Simulink

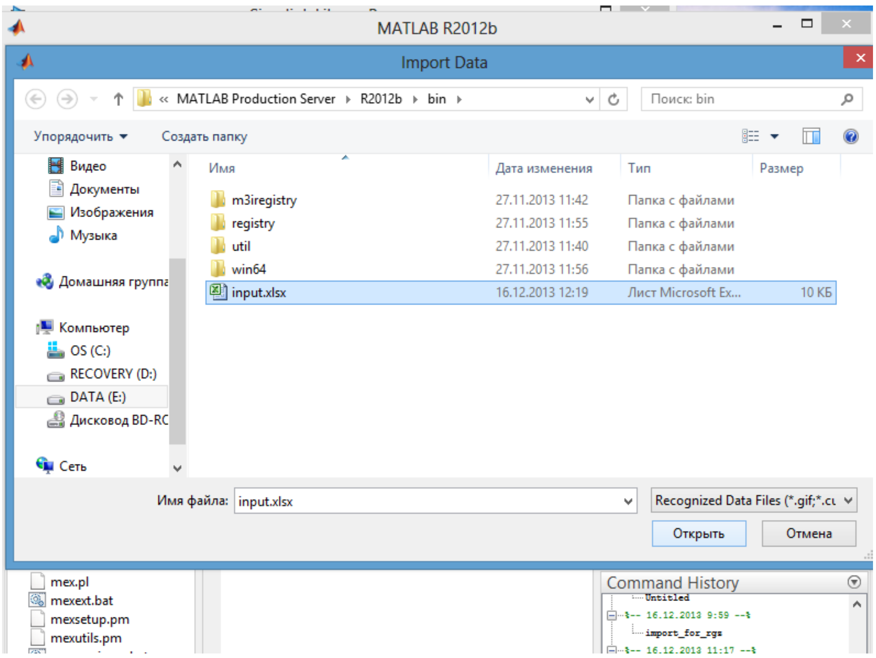


Рисунок 3.36 – Процесс создания модели в Simulink

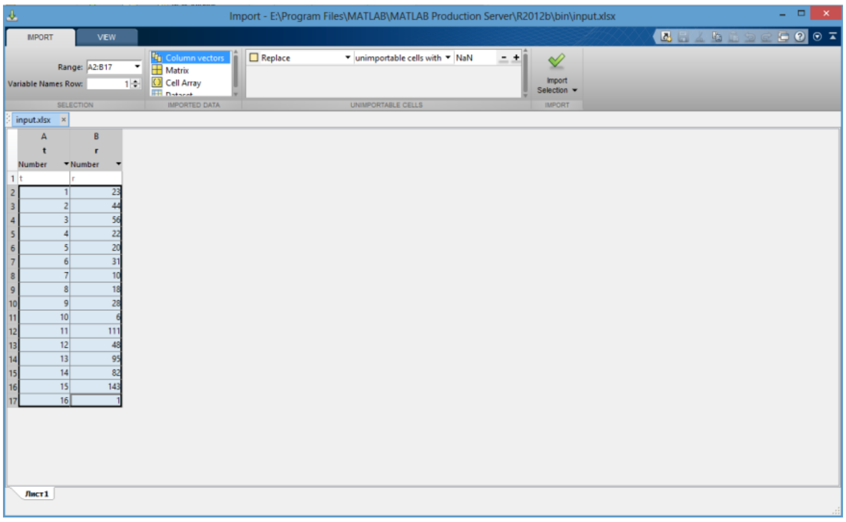


Рисунок 3.37 – Процесс создания модели в Simulink

4) после этого нажимаем «Import Selection», и выбираем один из трех вариантов импорта. «Import data» - позволяет нам выгрузить данные в рабочую область Matlab один раз, однако после завершения работы с приложением Matlab эти данные будут стерты, и нам придется выгружать их заново. «Generate Script» - создает специальный загрузочный файл, который будет находиться в рабочей папке Matlab и позволит выгружать данные всякий раз при запуске приложения. «Generate Function» - позволяет выгружать данные в виде функции (полезно при работе с большими массивами данных). В нашем случае наиболее удобно будет выбрать вариант «Generate Script», как показано на рисунке 3.38;

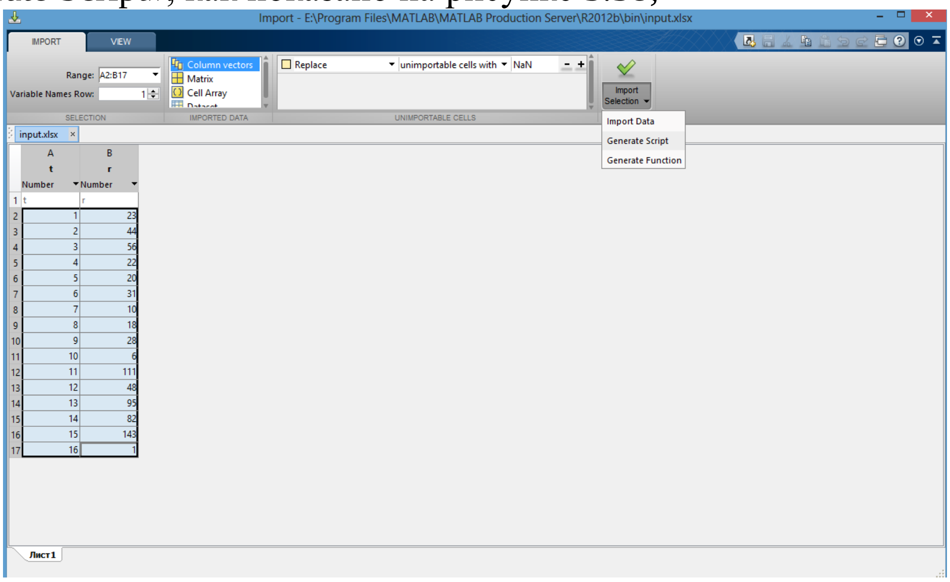


Рисунок 3.38 – Процесс создания модели в Simulink

5) после выбора вышеуказанного варианта, Matlab сгенерирует M-File нашего скрипта, который мы должны будем сохранить в рабочую папку Matlab под удобным для нас именем (файл имеет расширение .m), как показано на рисунке 3.39;

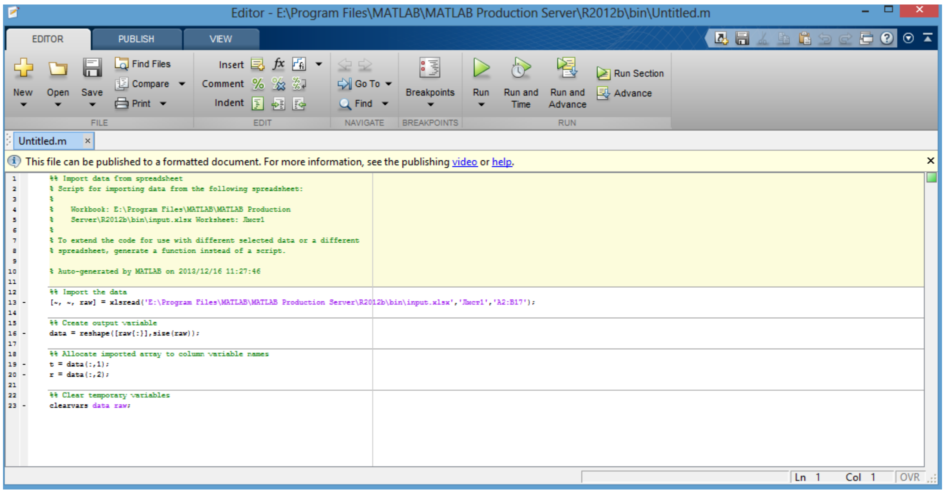


Рисунок 3.39 – Процесс создания модели в Simulink

6) затем мы можем запустить M-File скрипта, согласно рисунку 3.40;

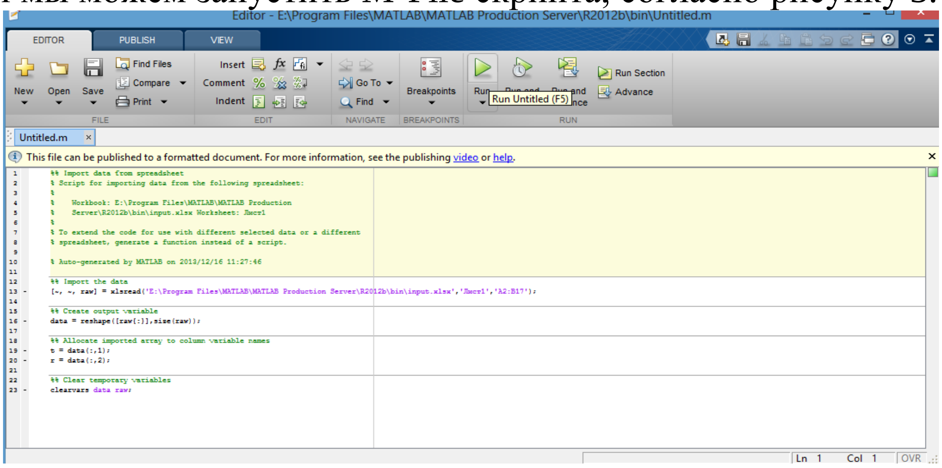


Рисунок 3.40 – Запуск M-File скрипта модели в Simulink

7. Теперь, когда необходимые для работы исходные данные загружены в Matlab, возвращаемся к настройке схемы. В верхнем окне мы можем выставить необходимое нам число рабочих тактов, по достижении которого Matlab прекратит работу со схемой (проще всего выставить количество тактов, равное длине вектора времени, загруженного нами в рабочую область Matlab). Пример настройки числа тактов приведен на рисунке 3.41.

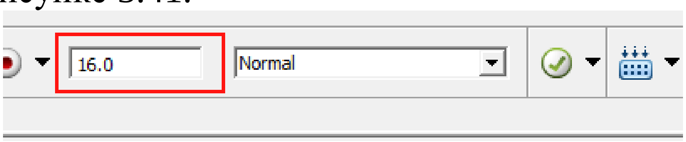


Рисунок 3.41 – Настройка числа тактов

8. После этого производим настройку блока Scope. Открываем меню настройки параметров (правый клик мыши и выбор пункта «Block Parameters» либо двойной щелчок левой кнопкой мыши). В любом месте появившегося экрана нажимаем правой кнопкой мыши и выбираем «Axes properties». В появившемся окне, показанном на рисунке 3.42, мы можем редактировать границы области значений для нашего графика, согласно рисунку 3.43.

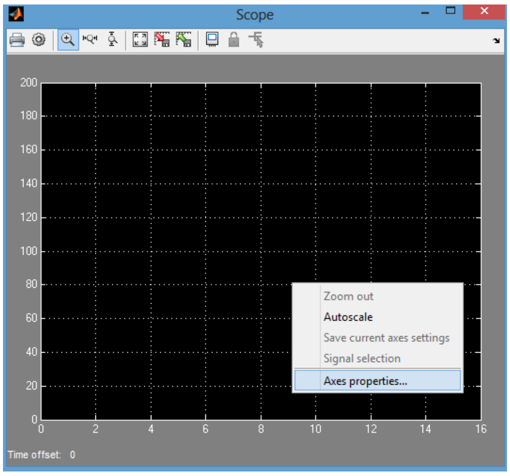


Рисунок 3.42 – Настройка блока Scope

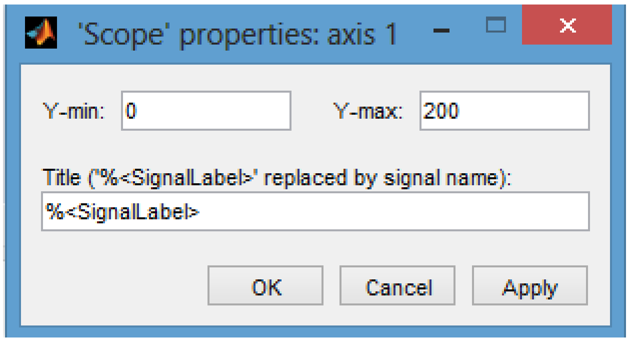


Рисунок 3.43 – Редактирование границы области значений графика

9. На этом настройка схемы завершена, и мы можем запустить ее, нажав на кнопку «Run», как показано на рисунке 3.44.

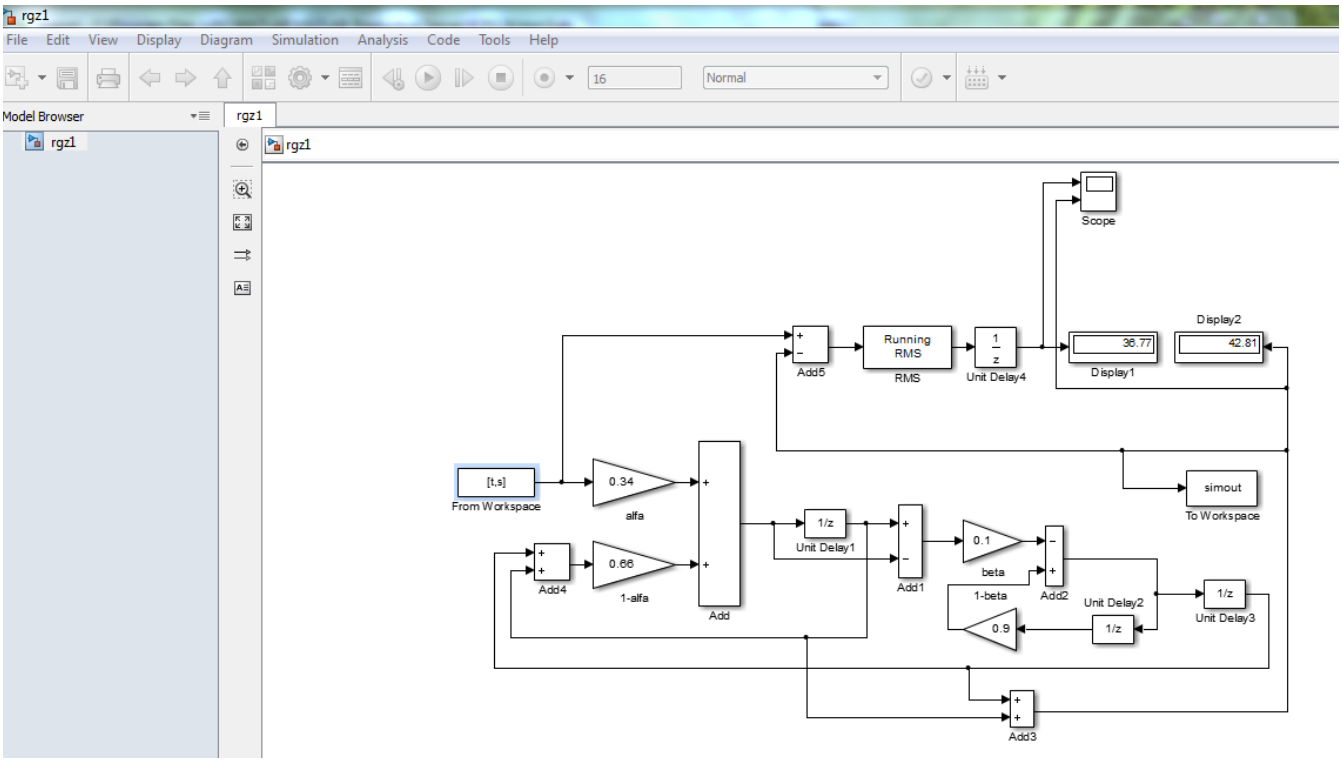


Рисунок 3.44 – Запуск полученной модели

Результаты работы представлены в блоке Scope, изображенном на рисунке 3.45.

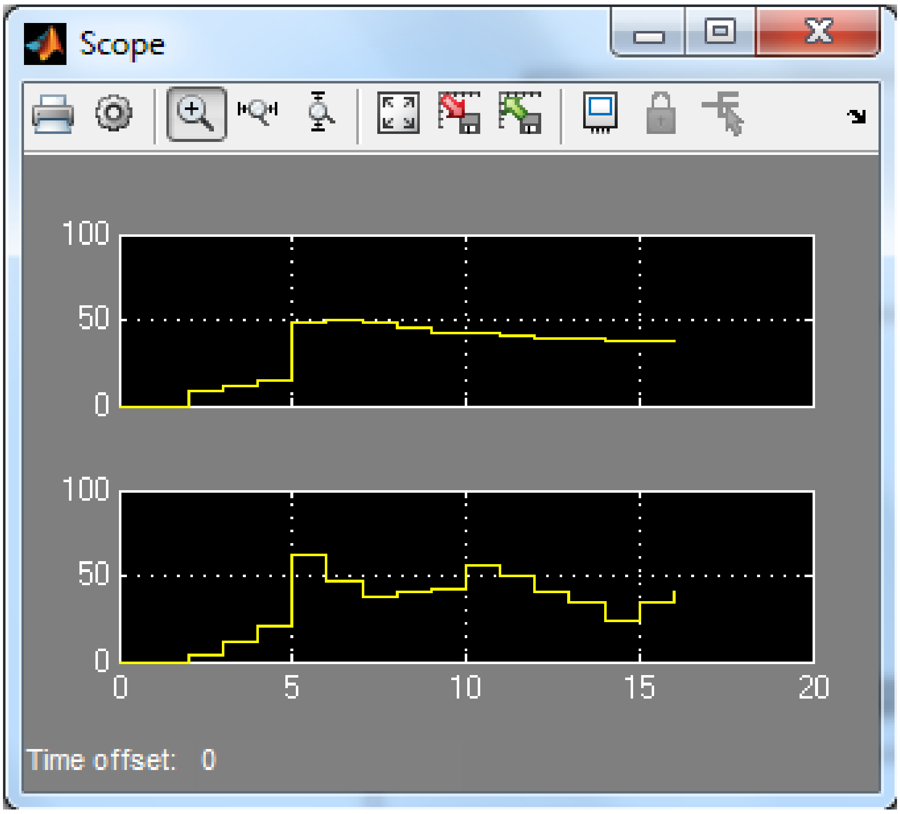


Рисунок 3.45 – Показания блока Scope

4 Литература, рекомендуемая для изучения

1 Вичугова, А.А. Инструментальные средства информационных систем: учебное пособие / А.А. Вичугова ; Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский Томский государственный университет», Министерство образования и науки Российской Федерации. – Томск : Издательство Томского политехнического университета, 2015. – 136 с. : ил., табл., схем. – Библиогр. в кн.. – ISBN 978-5-4387-0574-1. – Режим доступа: http://biblioclub.ru/index.php?page=book&id=442814

2 Инструментальные средства математического моделирования : учебное пособие / А.А. Золотарев, А.А. Бычков, Л.И. Золотарева, А.П. Корнюхин ; Министерство образования и науки Российской Федерации, Федеральное государственное автономное образовательное учреждение высшего профессионального образования «Южный федеральный университет". – Ростов-н/Д : Издательство Южного федерального университета, 2011. – 90 с. - библиогр. с. 88. – ISBN 978-5-9275-0887-7. – Режим доступа: http://biblioclub.ru/index.php?page=book&id=241127

3Плещинская И. Е. Интерактивные системы Scilab, Matlab, Mathcad: учебное пособие [Электронный ресурс] / Плещинская И. Е., Титов А. Н., Бадертдинова Е. Р. – Казань: КНИТУ, 2014.

4 Волков, В.Л. Моделирование процессов и систем в приборостроении. Учебное пособие для студентов технических специальностей дневной, вечерней и заочной форм обучения / В.Л. Волков. – Арзамас, АПИ НГТУ, 2008. – 143с. – Режим доступа: http://window.edu.ru/resource/326/60326/files/model2008_Posob.pdf

5 Афанасьев, В. Н. Анализ временных рядов и прогнозирование [Текст] : учебник для студентов высших учебных заведений, обучающихся по специальности 080601 "Статистика" и другим экономическим специальностям / В. Н. Афанасьев, М. М. Юзбашев. – М. : Финансы и статистика : ИНФРА-М, 2010. – 319 с. : ил. - Библиогр.: с. 301-302. – Прил.: с. 303-313. – ISBN 978-5-279-03400-0.

6 Бобков С.П. Моделирование систем: учеб. пособие / С.П. Бобков, Д.О. Бытев; Иван. гос. хим.-технол. ун-т. – Иваново, 2008 – 156 с. – Режим доступа:

http://main.isuct.ru/files/publ/PUBL_ALL/ivt/ivt2_18092008.pdf

7 Черных, И.В. Моделирование электротехнических устройств в MATLAB, SimPowerSystems и Simulink / И.В.Черных. – Издательство: "ДМК Пресс" , 2012. – 288 с. – ISBN: 9785940747369

8 Афанасьев, В. Н. Моделирование и прогнозирование временных рядов [Текст] : учеб.-метод. пособие / В. Н. Афанасьев, Т. В. Лебедева. – М. : Финансы и статистика, 2009. – 292 с. : ил. – Прил.: с. 251-286. – Библиогр.: с. 287. – ISBN 978-5-279-03402-4.

9 Терёхин, В.В. Моделирование в системе MATLAB: учебное пособие / В.В. Терёхин. – Новокузнецк: (Кемеровский государственный университет). Кузбассвузиздат, 2004. – 376с. – Режим доступа:
http://window.edu.ru/resource/199/56199/files/nkfi19.pdf

10Тимохин, А. Н. Моделирование систем управления с применением Matlab : учеб. пособие / А.Н. Тимохин, Ю.Д. Румянцев ; под ред. А.Н. Тимохина. – М. : ИНФРА-М, 2019. – 256 с. + Доп. материалы [Электронный ресурс; Режим доступа http://www.znanium.com]. – (Высшее образование: Бакалавриат). – www.dx.doi.org/10.12737/14347. – Режим доступа:
http://znanium.com/catalog/product/1004245

Приложение А

(обязательное)

Варианты заданий

Таблица А1– Варианты заданий

№ вар.	Задание 1	Задание 2
	Построить и исследовать имитационную модель	**Построить и исследовать модель динамической системы, описанную дифференциальным уравнением**
1	непрерывной системы контроля	$\frac{d^2y}{dx^2}+2n(1+\alpha y^3)\frac{dy}{dx}+k^2y=h\sin px,$
2	прогнозирования временных рядов методом Холта.	$\frac{d^2y}{dx^2}+2n\frac{dy}{dx}+k^2(y+\alpha y^2)=h\sin px,$
3	системы массового обслуживания с отказами	$\frac{d^2y}{dx^2}+2n\frac{dy}{dx}+k^2(y+\alpha y^3)=h\sin px,$
4	прогнозирования временных рядов методом скользящей средней	$\frac{d^2y}{dx^2}+2n(1+\alpha y^2)\frac{dy}{dx}+k^2y=h\sin px,$
5	прогнозирования временных рядов методом экспоненциального сглаживания	$\frac{d^2y}{dx^2}+2n(1+\alpha y^3)\frac{dy}{dx}+k^2(y+\beta y^2)=h\sin px,$
6	прогнозирования временных рядов методом Холта-Винтера (мультипликативная модель с удалением тренда)	$\frac{d^2y}{dx^2}+2n(1+\alpha y^3)\frac{dy}{dx}+k^2(y+\beta y^3)=h\sin px,$
7	прогнозирования временных рядов методом Холта-Винтера (аддитивная модель с удалением тренда)	$\frac{d^2y}{dx^2}+2n(1+\alpha y^3)\frac{dy}{dx}+k^2\sin y=h\sin px,$
8	прогнозирования временных рядов методом скользящей средней с использованием дискретного фильтра	$\frac{d^2y}{dx^2}+2n\cos y\frac{dy}{dx}+k^2(y+\beta y^2)=h\sin px,$
9	системы массового обслуживания с общей очередью	$\frac{d^2y}{dx^2}+2n\cos y\frac{dy}{dx}+k^2\sin(y+\beta y^2)=h\sin px,$

Продолжение таблицы А1

№ вар.	Задание 1	Задание 2
10	системы массового обслуживания с персональной очередью	$\frac{d^2y}{dx^2}+2n\left(\frac{dy}{dx}\right)^2+k^2(y+\beta y^2)=h\sin\ px,$
11	системы массового обслуживания с утилизацией заявок	$\frac{d^2y}{dx^2}+2n\left(\frac{dy}{dx}\right)^2+k^2\sin\ y=h\sin\ px,$
12	системы массового обслуживания с ограниченной очередью и обратной связью	$\frac{d^2y}{dx^2}+2n\left(\frac{dy}{dx}\right)^2+k^2(y+\beta y^3)=h\sin\ px,$
13	прогнозирования временных рядов методом Холта-Винтера (мультипликативная модель).	$\frac{d^2y}{dx^2}+2n\left(\frac{dy}{dx}\right)^2+k^2y=h\sin\ px,$
14	прогнозирования временных рядов методом Холта-Винтера (аддитивная модель)	$\frac{d^2y}{dx^2}+2n\left(\frac{dy}{dx}\right)^2+k^2(y+\beta y^2)=h\sin px,$
15	непрерывной системы контроля	$\frac{d^2y}{dx^2}+2n\left(\frac{dy}{dx}\right)^2+k^2(y+\beta y^2)=h\sin px,$
16	прогнозирования временных рядов методом Холта.	$\frac{d^2y}{dx^2}\quad 2n(1+\alpha y^3)\frac{dy}{dx}+k^2y=h\sin\ px,$
17	системы массового обслуживания с отказами	$\frac{d^2y}{dx^2}\quad 2n(1+\alpha y^3)\frac{dy}{dx}+k^2y=h\sin\ px,$
18	прогнозирования временных рядов методом скользящей средней	$\frac{d^2y}{dx^2}\quad 2n\left(\frac{dy}{dx}\right)^2+k^2(y+\beta y^2)=h\sin px,$
19	прогнозирования временных рядов методом экспоненциального сглаживания	$\frac{d^2y}{dx^2}\quad 2n\left(\frac{dy}{dx}\right)^2+k^2(y+\beta y^2)=h\sin px,$
20	прогнозирования временных рядов методом скользящей средней с использованием дискретного фильтра	$\frac{d^2y}{dx^2}+2n\left(\frac{dy}{dx}\right)^2+k^2\ln y=h\sin px.$