

**МИНИСТЕРСТВО ВЫСШЕГО И СРЕДНЕГО СПЕЦИАЛЬНОГО
ОБРАЗОВАНИЯ РЕСПУБЛИКА УЗБЕКИСТАН**

**ТАШКЕНТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ**

АВИАЦИОННЫЙ ФАКУЛЬТЕТ

На правах рукописи

Омонов Санжар Шухратович

**ДИССЕРТАЦИЯ
НА СОИСКАНИЕ СТЕПЕНИ МАГИСТРА**

на тему: **“Разработка виртуальной функциональной модели схемы
регулируемой индукционной муфты скольжения”**

Специальность: «5А310403 – Техническая эксплуатация авиационных
электросистем и пилотажно-навигационных комплексов»

Научный руководитель:

доц., к.т.н. Алиев А.Ш.

Ташкент – 2013г.

Аннотация

В данной работе на основе математического описания динамических процессов в реверсивной индукционной муфте скольжения предлагается структура и схема виртуальной модели РИМС, а также была разработана виртуальная лабораторная схема РИМС.

Для проверки правильности полученной виртуальной модели были проведены эксперименты.

Оглавление

Условные обозначения.	5
ВВЕДЕНИЕ	6
ГЛАВА 1. СИСТЕМА MATLAB – научно-исследовательская лабораторная программа.....	10
1.1 История появления MATLAB.....	10
1.2. Место MATLAB среди математических программ.....	11
1.3 Возможности, визуализация и графические средства.....	12
1.4. Средства программирования	14
ГЛАВА 2. РЕГУЛИРУЕМАЯ ИНДУКЦИОННАЯ МУФТА СКОЛЬЖЕНИЯ КАК ОБЪЕКТ ИССЛЕДОВАНИЯ	17
2.1. Конструкция регулируемой индукционной муфты скольжения.	17
2.2. Принцип действия РИМС.	18
2.3. Режимы работы РИМС.	20
2.3. Режимы работы регулируемой индукционной муфты скольжения.	24
ГЛАВА 3. МАТЕМАТИЧЕСКАЯ МОДЕЛЬ РЕГУЛИРУЕМОЙ ИНДУКЦИОННОЙ МУФТЫ СКОЛЬЖЕНИЯ МУФТЫ.....	29
3.1. Физическое содержание переменных.	29
3.2. Уравнения динамики РИМС.....	30
3.3. Выбор системы координат.	38
ГЛАВА 4. РАЗРАБОТКА ВИРТУАЛЬНОЙ ФУНКЦИОНАЛЬНОЙ МОДЕЛИ СХЕМЫ РЕГУЛИРУЕМОЙ ИНДУКЦИОННОЙ МУФТЫ СКОЛЬЖЕНИЯ	39
4.1 Пакет визуального программирования Simulink	39
4.2. Параметры модели для РИМС серии 4А	43
4.3. Обобщенная структурная модель регулируемой индукционной муфты скольжения.....	45
4.4. Разработка виртуальной функциональной модели регулируемой индукционной муфты скольжения.....	53
4.5 Результаты моделирования	64

ЗАКЛЮЧЕНИЕ	66
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	67
ПРИЛОЖЕНИЯ.....	70

Условные обозначения.

ЛА – летательный аппарат

ЭП – электропривод

ДПТ – двигатель постоянного тока

РИМС – реверсивная индукционная муфта скольжения

Ω_{10}^u – угловая частота тока, напряжения и магнитного поля обмотки индуктора

Ω_i – угловая частота (скорость вращения) индуктора

Ω_a – угловая частота (скорость вращения) якоря;

Ω_1^u – угловая частота магнитного поля якоря в зазоре

Ω_1^a – угловая частота магнитного поля индуктора в зазоре

E_{2k}, E_2 – ЭДС обмоток индуктора и якоря;

α_1^u – начальная угловая питающего напряжения в о.е.;

α_i, α_a – скорости вращения индуктора и якоря в о.е.;

$\beta = \alpha_1 s = \alpha_1 s_1 s_2$ – абсолютное скольжение якоря относительно магнитного поля в зазоре;

s_2, s_1 – относительное скольжение ротора 2 относительно магнитного поля в зазоре и относительное скольжение индуктора;

$\alpha = \alpha_1 - \alpha_1^p$ – угловая частота магнитного поля в о.е.;

ψ_{1k}, ψ_{2k} – потокосцепление k -ой фазы обмоток индуктора и якоря;

$\dot{U}_{1m}, \dot{I}_{1m}, \dot{I}_{2m}$ – амплитуды напряжения и токов обмоток индуктора якоря;

X_{1n}, X_{2n} – полные индуктивные сопротивления обмоток индуктора и якоря;

X_{1sn}, X_{2sn} – индуктивные сопротивления рассеивания индуктора и якоря;

X_{12n}, X_{21n} – индуктивные сопротивления взаимной индукции обмоток

Индуктора и якоря;

ВВЕДЕНИЕ

Развитие промышленности и сельского хозяйства неразрывно связано с ростом энерговооружённости, расширением номенклатуры машин и механизмов, оснащённых различного рода электромеханическими системами (ЭМС) и улучшением их характеристик. Поэтому всё большее распространение получают автономные системы электроприводов с широким диапазоном плавного регулирования частоты вращения, автономные системы электроснабжения, в том числе и с применением новых электрических машин (ЭМ) [1]. Эти системы применяются в сельском хозяйстве, промышленности, на летательных аппаратах и поездах, на самоходных, строительных и транспортных машинах. Быстрое развитие электроники и вычислительной техники и создание на этой основе различного регуляторов и статических преобразователей даёт возможность широко внедрять ЭМС на базе электрических машин переменного тока.

Работа ЭМ в таких системах существенно отличается от работы ЭМ общего назначения. Во-первых, обычно ЭМ работают при изменяющихся в заданных диапазонах или по заданным законам значений частот вращения, частот тока, напряжения. Во-вторых, очень часто ЭМ работает совместно с полупроводниковыми управляющими устройствами. В-третьих, переходные процессы ЭМ в таких системах носят не эпизодический, а непрерывный характер, вследствие чего их можно считать нормальным рабочим процессом, которым необходимо управлять с помощью регулируемых параметров, которые, как правило, связывают с электромагнитной и электромеханической инерцией ЭМ. Установление взаимосвязи между показателями переходного процесса и параметрами ЭМ позволяет произвести оптимизационные расчёты ЭМ, т.е. спроектировать электрическую машину, параметры которой будут удовлетворять заданным показателям динамических режимов.

Современные компьютерные технологии позволяют качественно изменить и существенно улучшить технологию изучения электрических машин, перевести её в виртуальную действительность, осуществить в этой виртуальной лаборатории исследования статических и динамических режимов работы электрических машин, их механических характеристик, условий пуска и технико-экономических показателей с получением количественных результатов.

Для грамотного использования компьютерных технологий при исследовании электрических машин необходимо хорошо знать и понимать физические процессы, протекающие в ЭМ; знать уравнения, описывающие работу электрической машины (ЭМ); уметь рассчитать параметры для построения математических моделей. Использование компьютерных технологий позволит расширить круг и глубину изучаемых вопросов, провести множество экспериментов с использованием виртуальных ЭМ, что позволяет установить взаимосвязи между показателями переходных процессов и параметрами машины.

Цель и задачи исследования.

Целью диссертационной работы является разработка виртуальной модели регулируемой индукционной муфты скольжения

Для достижения поставленной цели были поставлены задачи:

- определить математическую модель реверсивной индукционной муфты скольжения для динамического режима РИМС);
- разработать виртуальную модель РИМС для динамического режима;
- разработать схемы виртуальную модель РИМС и виртуальной лабораторной модели для исследования переходного режима;
- собрать схему виртуальной лабораторной модели на компьютере с целью проверки правильности разработанной виртуальной модели РИМС.

Общая методика исследования.

При разработке виртуальной модели РИМС для динамического режима использованы теория асинхронных электрических машин и теория векторно-комплексного анализа переходных процессов в электрических машинах и основы теории системы MATLAB

Научная новизна:

- разработана виртуальная модель РИМС для динамического режима и её схема;
- разработана и испытана компьютерная лабораторная схема виртуальной модели РИМС.

Практическая ценность: (где можно использовать)

Компьютерная лабораторная схема виртуальной модели РИМС позволяет:

- исследовать возможные динамические режимы РИМС (пуск, останов, торможение, сброс и наброс нагрузки . . .)
- выявлять параметры РИМС, которые будут удовлетворять заданным показателям динамических режимов РИМС.

Основные защищаемые положения.

1. Виртуальная модель РИМС для динамического режима и её схема;
2. Схема лабораторной схема виртуальная модели РИМС;

Апробация работы и публикации.

1. Разработка схемы функционального блока расчёта токов и потокосцеплений регулируемой индукционной муфты скольжения, журнал «Техника юлдузлари», в печати 2012 г.;
2. Схемы функциональных блоков расчёта момента и угловой частоты якоря регулируемой индукционной муфты скольжения, журнал «Техника юлдузлари», в печати 2012 г.;
3. Разработка схемы виртуальной функциональной модели регулируемой индукционной муфты скольжения, Республиканская научно-

техническая конференция «Актуальные научно-технические проблемы авиационной отрасли», ТГТУ, 2013 г.

ГЛАВА 1. СИСТЕМА MATLAB – научно-исследовательская лабораторная программа

1.1 История появления MATLAB

Система MATLAB [2] разработана специалистами компании MathWorkInc. (г. Нейтик, штат Массачусетс, США). Хотя впервые эта система начала использоваться в конце 1970-х годов, широкое распространение она получила в конце 80-х, в особенности после появления на рынке версии 4.0. Последние версии MATLAB -это системы, которые содержат множество процедур и функций, необходимых инженеру и научному работнику для осуществления сложных численных расчетов, моделирования технических и физических систем и оформления результатов этих расчетов. MATLAB (сокращение от MATrixLABoratory - матричная лаборатория) представляет собой интерактивную систему, предназначенную для выполнения инженерных и научных расчетов и ориентированную на работу с массивами данных.

В последние годы в научных и инженерно-технических кругах получила широкое распространение система MATLAB. Более того, в настоящее время она принята в качестве официального средства оформления инженерной документации и научных публикаций. Система MATLAB специально создана для проведения именно инженерных расчетов: математический аппарат, который используется в ней, предельно приближен к современному математическому аппарату инженера и ученого. Функциональные зависимости здесь организованы в форме, которую требует именно инженерная документация.

1.2. Место MATLAB среди математических программ

У системы MATLAB есть схожие черты с программами MathCad и ElectronicsWorkbench. Так с MathCad её роднит то, что в обеих системах имеются широкие возможности по выполнению вычислений, производимых с матрицами, векторами и комплексными числами, а также графическое представление полученных результатов. Отличительной чертой является входной язык, максимально приближенный к обычному математическому языку.

А с ElectronicsWorkbench общим является возможность создания моделей как отдельных объектов так и систем, путём поблочного моделирования и с помощью специальных блоков наблюдать протекающие процессы в модели.

1.3 Возможности, визуализация и графические средства

Основной объект системы MATLAB - прямоугольный числовой массив (матрица), в котором допускается применение комплексных элементов. Использование матриц не требует явного указания их размеров.

Система MATLAB обеспечивает выполнение операций с векторами и матрицами даже в режиме непосредственных вычислений. Ею можно пользоваться как мощнейшим калькулятором, в котором наряду с обычными арифметическими и алгебраическими действиями могут использоваться такие сложные операции, как обращение матрицы, вычисление ее собственных значений и векторов, решение систем линейных алгебраических уравнений и много других. Характерной особенностью системы является ее открытость, то есть возможность ее модификации и адаптации к конкретным задачам пользователя.

Привлекательной особенностью системы MATLAB является наличие встроенной матричной и комплексной арифметики. Система поддерживает выполнение операций с векторами, матрицами и массивами данных, реализует сингулярное и спектральное разложение, расчет ранга и чисел обусловленности матриц, поддерживает работу с алгебраическими полиномами, решение нелинейных уравнений и задач оптимизации, интегрирование функций в квадратурах, численное интегрирование дифференциальных и разностных уравнений, построение различных графиков, трехмерных поверхностей и линий уровня.

MATLAB предоставляет широкие возможности для работы с сигналами, для расчета и проектирования аналоговых и цифровых фильтров, включая построение их частотных, импульсных и переходных характеристик. Имеются в системе и средства выполнения спектрального анализа и синтеза, в частности реализации прямого и обратного преобразования Фурье. Благодаря этому ее довольно удобно использовать при проектировании электронных устройств.

Одной из наиболее привлекательных особенностей системы MATLAB является наличие в ней наглядного и эффективного средства составления программных моделей - пакета визуального программирования Simulink.

Пакет Simulink позволяет осуществлять исследование (моделирование во времени) поведения динамических линейных и нелинейных систем, причем составление «программы» и ввод характеристик систем можно производить в диалоговом режиме, путем сборки на экране схемы соединений элементарных звеньев. В результате такой сборки получается модель системы, называемая S-моделью. В качестве «кирпичиков» при построении S-модели применяются визуальные блоки (модули), которые сохраняются в библиотеках Simulink.

S-модель может иметь иерархическую структуру, то есть состоять из моделей более низкого уровня, причем количество уровней иерархии практически не ограничено.

1.4. Средства программирования

Система обеспечивает возможность обращения к программам, которые написаны на языках FORTRAN, C и C++.

Система MATLAB использует собственный М-язык, который сочетает в себе положительные свойства различных известных языков программирования высокого уровня. С языком BASIC систему MATLAB роднит то, что она представляет собой интерпретатор (осуществляет пооператорное компилирование и выполнение программы, не образуя отдельного исполняемого файла), М-язык имеет незначительное количество операторов, в нем отсутствует необходимость объявлять типы и размеры переменных. От языка Pascal система MATLAB позаимствовала объектно-ориентированную направленность, то есть такое построение языка, которое обеспечивает образование новых типов вычислительных объектов на основе типов объектов, уже существующих в языке. Новые типы объектов (в MATLAB они называются классами) могут иметь собственные процедуры их преобразования (они определяют методы этого класса), причем новые процедуры могут быть вызваны с помощью обычных знаков арифметических операций и некоторых специальных знаков, которые применяются в математике.

Принципы сохранения значений переменных в MATLAB наиболее близки к тем, которые присущи языку FORTRAN, а именно: все переменные являются локальными - действуют лишь в границах той программной единицы (процедуры, функции или главной, управляющей программы), где им присвоены некоторые конкретные значения. При переходе к выполнению другой программной единицы, значения переменных предыдущей программной единицы либо теряются (в случае, если выполненная программная единица представляет собой процедуру или функцию), либо становятся недостижимыми (если выполненная программа является управляющей). В отличие от языков BASIC и Pascal, в

языке MATLAB нет глобальных переменных, действие которых распространялось бы на все программные единицы. Но при этом язык MATLAB обладает возможностью, которая отсутствует в других языках. Интерпретатор MATLAB позволяет в одном и том же сеансе работы выполнять несколько самостоятельных программ, причем все переменные, используемые в этих программах, являются для них общими и образуют единое рабочее пространство. Это дает возможность более рационально организовывать сложные (громоздкие) вычисления по типу оверлейных структур.

Язык программирования системы MATLAB весьма прост, он содержит лишь несколько десятков операторов; незначительное количество операторов здесь компенсируется большим числом процедур и функций, содержание которых понятно пользователю, имеющему соответствующую математическую и инженерную подготовку.

В отличие от большинства математических систем, MATLAB является открытой системой: практически все ее процедуры и функции доступны не только для использования, но и для модификации. Почти все вычислительные возможности системы можно применять в режиме чрезвычайно мощного научного калькулятора, а также составлять собственные программы, предназначенные для многократного применения; это делает MATLAB незаменимым средством проведения научных исследований. По скорости выполнения задач MATLAB опережает многие другие подобные системы. Все эти особенности делают ее весьма привлекательной для использования.

С системой MATLAB поставляются свыше ста подробно прокомментированных М-файлов, которые содержат демонстрационные примеры и определения новых операторов и функций. Наличие этих примеров и возможность работать в режиме непосредственных

вычислений значительно облегчают изучение системы пользователями, заинтересованными в применении математических расчетов.

ГЛАВА 2. РЕГУЛИРУЕМАЯ ИНДУКЦИОННАЯ МУФТА СКОЛЬЖЕНИЯ КАК ОБЪЕКТ ИССЛЕДОВАНИЯ

2.1. Конструкция регулируемой индукционной муфты скольжения.

Регулируемая индукционная муфта скольжения (РИМС) [3] относится к электрической машине с двумя степенями свободы. Принципиальная конструкция РИМС показана на рис.2.1. Основными элементами конструкции муфты являются: индуктор 1, вращающийся от вала ходового двигателя (ХД) с угловой частотой Ω_i ; вращающийся с угловой скоростью Ω_a якорь 2, к валу которого подсоединён исполнительный механизм (ИМ); корпус 3, удерживающий вращающиеся части муфты; щёточно-контактное устройство 4; симметричный трёхфазный источник электрического тока. Необходимо отметить внешний индуктор конструктивно может быть и внутренним, тогда якорь будет внешним.

Особенностью РИМС является то, что в пазы сердечников индуктора и якоря уложены соответственно трёхфазная симметричная обмотка и короткозамкнутая обмотка в виде беличьей клетки. К фазам обмотки индуктора подключен через щеточно-контактное устройства источник электрического тока.

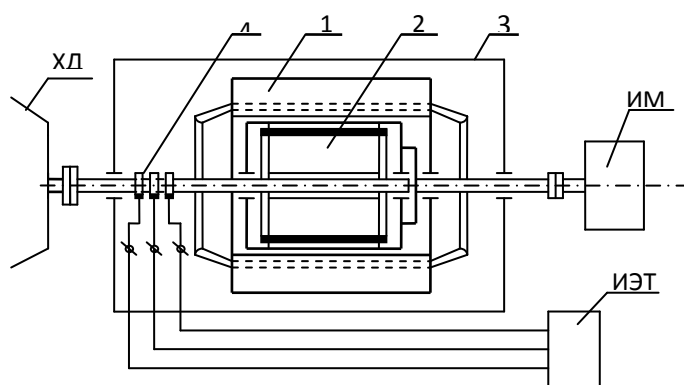


Рис.2.1. Конструкция регулируемой индукционной муфты скольжения.

2.2. Принцип действия РИМС.

Для объяснения принципа действия индукционной муфты скольжения введём следующие понятия[4].

Магнитное поле индуктора при неподвижном индукторе – это магнитное поле Φ_{10}'' , создаваемое фазными токами обмотки индуктора I_{10}'' и вращающееся с угловой частотой Ω_{10}'' . Величина, угловая частота и направление вращения магнитного поля в зазоре определяются соответственно величиной напряжения U_1'' , частотой тока источника f_u , чередованием фаз токов.

Индукторное магнитное поле в зазоре при вращающемся индукторе – это создаваемое фазными токами обмотки индуктора I_1'' магнитное поле Φ_1'' , величина которого определяется величиной напряжения U_1'' , а угловая частота Ω_1'' и направление которых зависят от соотношения величин и направлений угловых частот соответственно индуктора Ω_i и магнитного поля Ω_{10}'' , созданного фазными токами обмотки индуктора.

Магнитное поле якоря при неподвижном якоре – это магнитное поле, величина Φ_{10}^a и угловая частота Ω_{10}^a которого зависят от величины, угловой частоты и направления индукторного магнитного поля в зазоре.

Якорное магнитное поле в зазоре – это магнитное поле при вращающемся якоре, величина Φ_{10}^a , угловая частота Ω_{10}^a и направление вращения которого зависят соответственно от величины магнитного поля якоря и от соотношения угловых частот якоря Ω_a и магнитного поля якоря Ω_{10}^a , созданного фазными токами обмотки якоря I_1^a .

Работа регулируемой индукционной муфты скольжения основывается на принципе электромагнитного взаимодействия между вращающимся индукторным магнитным полем в зазоре и токами, наводимыми в обмотке

якоря при пересечении её проводников вращающимся индукторным магнитным полем в зазоре.

Таким образом, работа муфты по своей физической сущности подобна работе трансформатора, причём индуктор можно рассматривать как первичную обмотку, а якорь – как вторичную. Особенностью является то, что индуктор и якорь вращаются соответственно со скоростью вращения Ω_i и Ω_a .

При синусоидальном характере вращающегося магнитного поля индуктора и якоря их скорости вращения определяются как

$$\begin{aligned}\Omega_{10}^u &= 2\pi \cdot f_u / p_u, \\ \Omega_{10}^a &= 2\pi \cdot f_a / p_a\end{aligned}\tag{2.1}$$

где f_u – частота тока в обмотке индуктора,

f_a – частота тока в обмотке якоря,

p_u, p_a – число пар полюсов соответственно индуктора и якоря.

Электромагнитное взаимодействие между обеими частями муфты возможно только при наличии разностей между угловыми частотами индукторного магнитного поля в зазоре Ω_{10}^u и якоря Ω_a . В противном случае муфта при $\Omega_{10}^u = \Omega_a$ не работает вследствие отсутствия ЭДС в якорной обмотке.

2.3. Режимы работы РИМС.

Рассмотрим возможные установившиеся режимы работы муфты, при этом за положительное направление вращения примем направление вращения магнитного поля индуктора, созданного фазными токами обмотки индуктора:

- согласное вращение индуктора и магнитного поля индуктора $\Omega_i \uparrow\uparrow \Omega_{10}''$;
- встречное вращение при угловой частоте индуктора, меньшей угловой частоты магнитного поля индуктора $\Omega_{10}'' > \Omega_i$, $\Omega_a > 0$, $\Omega_i \downarrow\uparrow \Omega_{10}''$;
- встречное вращение при угловой частоте индуктора, большей угловой частоты магнитного поля индуктора $\Omega_{10}'' < \Omega_i$, $\Omega_a < 0$, $\Omega_i \uparrow\downarrow \Omega_{10}''$;
- угловые частоты индуктора и магнитного поля индуктора равны $\Omega_{10}'' = \Omega_i$.

Определим угловые частоты индукторного магнитного поля в зазоре Ω_1'' и скольжения якоря и индуктора относительно магнитного поля в зазоре (s_a, s_i):

- при согласном вращении ($\Omega_i \uparrow\uparrow \Omega_{10}''$)

$$\begin{aligned}\Omega_1'' &= \Omega_{10}'' + \Omega_i, \\ s_a &= \frac{\Omega_1'' - \Omega_a}{\Omega_1''} = 1 - \frac{\Omega_a}{\Omega_1'' + \Omega_i}, \\ s_i &= \frac{\Omega_1'' - \Omega_i}{\Omega_1''} = 1 - \frac{\Omega_{10}'' + \Omega_i - \Omega_i}{\Omega_1''} = 1 - \frac{\Omega_{10}''}{\Omega_1''},\end{aligned}\tag{2.2}$$

- при встречное вращение для случая $\Omega_i \downarrow\uparrow \Omega_{10}''$, $\Omega_{10}'' > \Omega_i$, $\Omega_a > 0$

$$\Omega_1'' = \Omega_{10}'' - \Omega_i,$$

$$s_a = \frac{\Omega_1'' - \Omega_a}{\Omega_1''} = 1 - \frac{\Omega_a}{\Omega_{10}'' - \Omega_i}, \quad (2.3)$$

$$s_i = \frac{\Omega_1'' + \Omega_i}{\Omega_1''} = \frac{(\Omega_{10}'' - \Omega_i) + \Omega_i}{\Omega_1''} = \frac{\Omega_{10}''}{\Omega_{10}'' - \Omega_i}$$

– при встречное вращение для случая $\Omega_i \uparrow \downarrow \Omega_{10}'', \Omega_{10}'' < \Omega_i, \Omega_a < 0$

$$\Omega_1'' = -(\Omega_i - \Omega_{10}''),$$

$$s_a = \frac{-(\Omega_i - \Omega_{10}'') - (-\Omega_a)}{-(\Omega_i - \Omega_{10}'')} = -\left(1 - \frac{\Omega_a}{\Omega_i - \Omega_{10}''}\right) \quad (2.4)$$

$$s_i = \frac{-(\Omega_i - \Omega_{10}'') - (-\Omega_i)}{-(\Omega_i - \Omega_1'')} = -\left(1 - \frac{\Omega_i}{\Omega_i - \Omega_{10}''}\right)$$

Как видим из (2.4) и (2.4), для встречного вращения индуктора и магнитного поля, созданного обмоткой индуктора индукторное магнитное поле в зазоре меняет направление вращения, вследствие чего якорь также меняет направление вращения.

Для случая $\Omega_1'' = \Omega_i$ индукторное магнитное поле в зазоре неподвижно в пространстве и якорь находится в заторможенном состоянии.

Рассмотрим, соблюдается ли для РИМС третий закон электромеханики, гласящий о том, что электромеханическое преобразование энергии осуществляется полями, неподвижными относительно друг друга? Одновременно выясним, с какой угловой частотой якорное магнитное поле в зазоре пересекает проводники обмотки индуктора?

Угловая частота токов в обмотке якоря при согласном равна

$$\Omega_1^a = \Omega_{10}^u - \Omega_a ,$$

а при встречном вращении

$$\Omega_1^a = ((\Omega_1^u - \Omega_i) \mp \Omega_a) = (\Omega_{10}^u \mp \Omega_a) .$$

где « \rightarrow » для случая $\Omega_1^u > \Omega_i$ и « $+$ » – для случая $\Omega_1^u < \Omega_i$

Тогда угловая частота якорного магнитного поля в зазоре равна
– при согласном и встречном ($\Omega_1^u > \Omega_i$) вращениях

$$\Omega_{10}^a = \Omega_1^a + \Omega_a = \Omega_{10}^u - \Omega_a + \Omega_a = \Omega_{10}^u , \quad (2.8),$$

– при встречном вращении $\Omega_1^u < \Omega_i, \Omega_a < 0$

$$\Omega_{10}^a = \Omega_1^a - \Omega_a = -(\Omega_u - \Omega_1^u - \Omega_a) - \Omega_a = -(\Omega_u - \Omega_1^u) = -\Omega_{10}^u . \quad (2.9)$$

Определим угловую частоту пересечения проводников обмотки индуктора якорным магнитным полем в зазоре для всех случаев.

$$\Omega_{10}^a - \Omega_i = \Omega_{10}^u - \Omega_i = \Omega_1^u + \Omega_i - \Omega_i = \Omega_1^u \text{ при } \Omega_i \uparrow \uparrow \Omega_1^u ,$$

$$\Omega_{10}^a - (-\Omega_u) = \Omega_{10}^u + \Omega_u = \Omega_1^u - \Omega_u + \Omega_u = \Omega_1^u \text{ при } \Omega_i \downarrow \uparrow \Omega_1^u, \Omega_1^u > \Omega_i$$

$$\Omega_{10}^a - (-\Omega_u) = \Omega_{10}^u + \Omega_u = -(\Omega_u - \Omega_1^u) + \Omega_u = \Omega_1^u \text{ при } \Omega_1^u \downarrow \uparrow \Omega_i, \Omega_1^u < \Omega_i$$

Таким образом, третий закон электромеханики соблюдается, т.к. магнитные поля индуктора и якоря в зазоре в пространстве неподвижны относительно друг друга, и якорное магнитное поле в зазоре пересекает

проводники обмотки индуктора с угловой частотой, равной угловой частоте тока обмотки индуктора.

2.3. Режимы работы регулируемой индукционной муфты скольжения.

Рассмотрим режимы работы РИМС при:

- согласном вращении индуктора и магнитного поля индуктора;
- встречном вращении индуктора и магнитного поля индуктора;
- торможении муфты.

2.3.1. Режим согласного вращения индуктора и магнитного поля индуктора. Направления магнитного поля, тока, электромагнитной силы и угловая частота соответственно индуктора и якоря показаны на рис. 2.2.

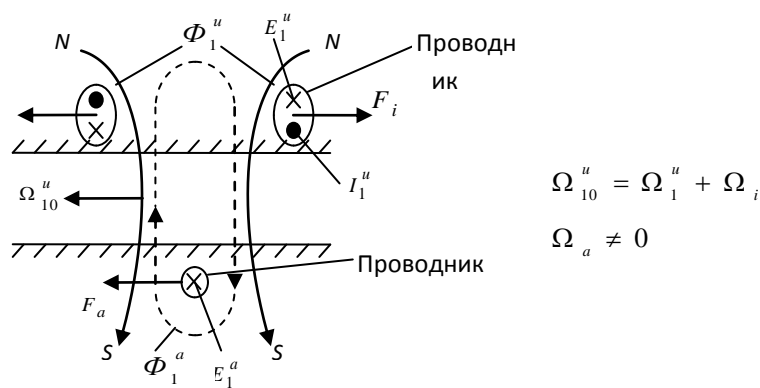


Рис.2.2. Работа РИМС в режиме муфты.

Как видно из рис.2.2, магнитное поле индуктора набегает справа на проводник якоря. Пересекая его, наводит в нём ЭДС (E_1^a), направление которой определяем по правилу правой руки с учётом условного встречного движения проводника магнитному полю индуктора. Так как обмотка якоря накоротко замкнута, то в ней протекает ток, активная составляющая которого имеет направление ЭДС. Применяя правило левой руки, определяют направление действия электромагнитной силы (F_a). Данная сила создаёт электромагнитный момент, вращающий якорь. Ток якоря создаёт магнитное поле якоря. С другой стороны, справа относительно проводника якорной обмотки плотность магнитных силовых

линий больше, чем слева. Поэтому электромагнитная сила F_a , приложенная к проводнику якоря, действует в направлении движения индукторного магнитного поля в зазоре Ω_{10}'' , заставляя якорь вращаться в том же направлении. Электромагнитный момент, приложенный к якорю, уравнивает момент сопротивления нагрузки. Плотность же магнитных силовых линий слева относительно левого проводника обмотки индуктора больше, чем справа. Создаваемая сила F_u от взаимодействия якорного поля с током обмотки индуктора действует против вращения индуктора. Следовательно, механический момент ХД уравнивает электромагнитный момент от силы F_u . С другой стороны, якорное магнитное поле в зазоре Φ_{10}^a , пересекая проводник обмотки индуктора, наводит в нём ЭДС E_1'' . направление которого совпадает с направлением активной составляющей тока проводника обмотки индуктора.

В этом режиме мощность источника электрического тока идёт на создание электромагнитного поля в зазоре для образования электромагнитного момента, а мощность ходового двигателя расходуется на создание механического момента, уравнивающего электромагнитный момент, приложенный к индуктору. Такой режим будем называть режимом *муфты*.

2.3.2. Режим встречного вращения индуктора и магнитного поля индуктора в условии $\Omega_1'' > \Omega_i$. Картина магнитных полей, направления токов, электромагнитных сил и угловых частот индуктора и якоря даны на рис. 2.3.

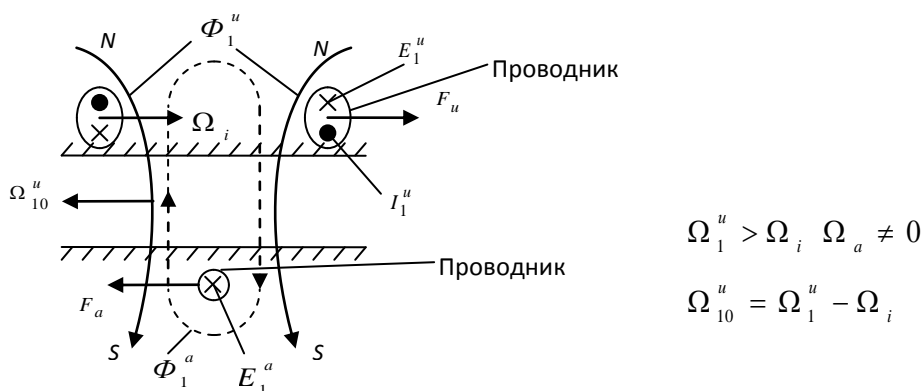


Рис.2.3. Работа РИМС в режиме двойного вращения.

Магнитное поле индуктора в зазоре вращается с угловой частотой $\Omega_{10}^u = \Omega_1^u - \Omega_i$ в противоположном направлении вращения индуктора. Сила F_a , возникающая от взаимодействия токов якоря с вращающимся индукторным магнитным полем в зазоре, действует на проводники якоря в направлении вращения индукторного магнитного поля в зазоре. Она создаёт вращающий момент, действующий в ту же сторону. Якорь вращается в направлении вращения магнитного поля индуктора в зазоре. При взаимодействии тока индуктора с вращающимся якорным магнитным полем в зазоре также создаётся сила F_u , направление действия которой противоположно действию силы F_a , приложенной к проводникам якоря. Поэтому при условии $\Omega_1^u > \Omega_i$ приводной двигатель разгружается (если двигатель электрический, то он может перейти в генераторный режим). Как видим, мощность электрического источника расходуется на раскрутку якоря и индуктора. Этот режим назовём режимом *двойного вращения*.

2.3.3. Режим встречного вращения индуктора и магнитного поля индуктора при условии $\Omega_1^u < \Omega_i$. Картина и направления магнитных полей, токов,

электромагнитных сил и угловых частот индуктора и якоря показаны на рис. 2.4.

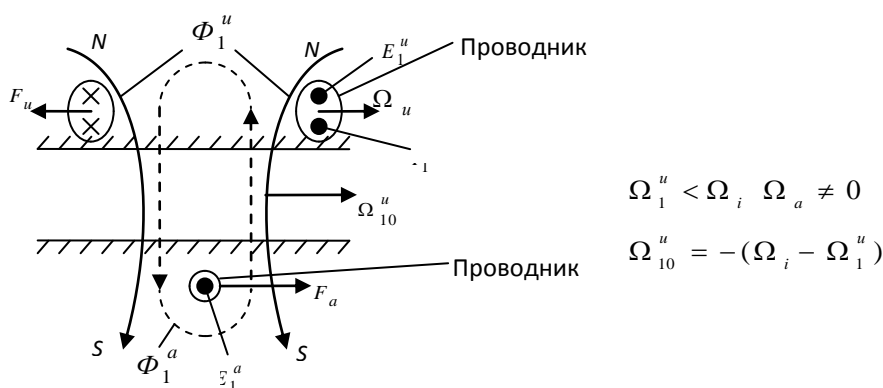


Рис.2.4. Работы РИМС в режиме муфты-генератора.

Индукторное магнитное поле в зазоре вращается с угловой частотой $\Omega_{10}^u = -(\Omega_i - \Omega_1^u)$ согласно с направлением вращения индуктора. От взаимодействия токов якоря с вращающимся индукторным магнитным полем в зазоре возникает сила F_a , действующая на проводники якоря в направлении вращения индуктора. Сила создаёт электромагнитный момент, вращающий якорь в ту же сторону. При взаимодействии токов индуктора с вращающимся магнитным полем якоря в зазоре возникает такая же сила F_u , действующая на проводники индуктора в направлении противоположном движению индуктора. Создаваемый её момент действует встречно моменту приводного двигателя, т.е. его загружает. С другой стороны, магнитный поток якоря в зазоре вращается в противоположном направлении вращения индуктора. Этот поток наводит ЭДС в обмотке индуктора, которая сдвинута относительно ЭДС источника на 180° . При превышении ЭДС источника часть энергии будет передаваться от приводного двигателя в электрический источник.

2.3.4. Для случая $\Omega_1'' = \Omega_i$ индукторное магнитное поле в зазоре неподвижно в пространстве, возникает тормозной режим муфты, аналогичный режиму асинхронного двигателя при подаче на обмотку статора постоянного тока. При всякой попытке якоря изменить положение возникает электромагнитный момент, препятствующий этому изменению.

ГЛАВА 3. МАТЕМАТИЧЕСКАЯ МОДЕЛЬ РЕГУЛИРУЕМОЙ ИНДУКЦИОННОЙ МУФТЫ СКОЛЬЖЕНИЯ МУФТЫ

3.1. Физическое содержание переменных.

Регулируемая индукционная муфта скольжения (РИМС) – это электрическая машина, у которой в индукторе, установленном на ведущем вале, уложена распределённая m_u -фазная симметричная обмотка, а в якоре, установленном на ведомом вале, уложена m_a -фазная симметричная короткозамкнутой обмоткой (демпферная обмотка).

Особенностью муфты является то, что направление и величина скорости вращения магнитного поля в зазоре Ω зависит от направления и величины скорости вращения магнитного поля, созданного токами обмотки индуктора Ω_1'' , относительно вращения самого индуктора Ω_i . Якорь, как известно, вращается в направлении вращения магнитного поля зазора. Благодаря этой особенности муфта обладает рядом свойств.

Возможны два варианта: согласное и встречное вращение индуктора и созданного им вращающегося магнитного поля. Магнитное поле в зазоре Ω при согласном вращении равно $\Omega = \Omega_1'' + \Omega_i$, при встречном вращении равно $\Omega = \Omega_1'' - \Omega_i$ для $\Omega_1'' > \Omega_i$ и $\Omega = -(\Omega_i - \Omega_1'')$ для $\Omega_1'' < \Omega_i$, а для $\Omega_1'' = \Omega_i$ – режим заторможенного якоря.

Рассмотрим РИМС, имеющую одну трёхфазную симметричную распределённую обмотку в индукторе [$m_u = 3$, $\alpha_{1k}'' = \alpha_{10} + (k - 1)2\pi / 3$] и демпферную обмотку в якоре. Демпферную обмотку представим в виде двух эквивалентных обмоток, магнитные оси которых взаимно перпендикулярны.

$$[N = 2, m_{1a} = m_{2a} = 1, \alpha_1^a = 0, \alpha_2^a = \pi / 2].$$

Эквивалентные обмотки демпфера накоротко замкнуты [$U_1^a = U_2^a = 0$].

3.2. Уравнения динамики РИМС.

Рассмотрим физическое содержание преобразований переменных. Представим в поперечной пространственной плоскости муфты (рис.3.1) фазные токи (i_{1k}^u, i_{1k}^a) , напряжения (u_{1k}^u, u_{1k}^a) и потокосцепления $(\psi_{1k}^u, \psi_{1k}^a)$ индуктора и якоря векторными величинами, модули которых определяются мгновенными значениями токов, напряжений и потокосцеплений, а направления – магнитными осями фаз обмоток индуктора и якоря

$$\alpha_{1k}^u = \alpha_{01}^u + (1 - k)2\pi / m_u, \quad \alpha_{1k}^a = \alpha_{01}^a + (1 - k)2\pi / m_a \quad (3.1)$$

Следовательно, для векторов токов, напряжений и потокосцеплений фаз обмоток индуктора и якоря будем иметь:

$$\begin{aligned} \bar{i}_{1k}^u &= i_{1k}^u e^{j\alpha_{1k}^u}, \quad \bar{u}_{1k}^u = u_{1k}^u e^{j\alpha_{1k}^u}, \quad \bar{\psi}_{1k}^u = \psi_{1k}^u e^{j\alpha_{1k}^u}, \\ \bar{i}_{1k}^a &= i_{1k}^a e^{j\alpha_{1k}^a}, \quad \bar{u}_{1k}^a = u_{1k}^a e^{j\alpha_{1k}^a}, \quad \bar{\psi}_{1k}^a = \psi_{1k}^a e^{j\alpha_{1k}^a}, \end{aligned} \quad (3.2)$$

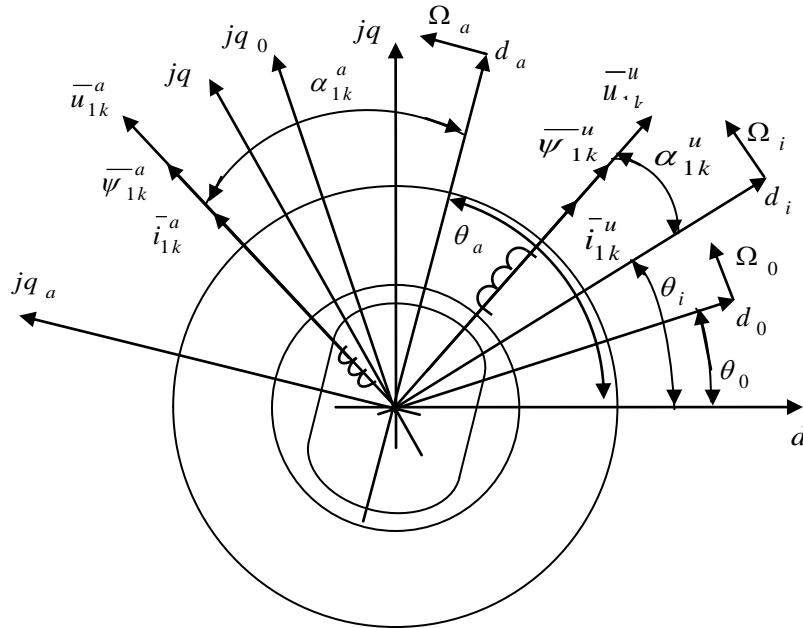


Рис.3.1. Векторные диаграммы напряжений, токов и потокосцеплений в координатных системах $d, jq, d_0, jq_0, d_i, jq_i, d_a, jq_a$

Для фаз обмоток индуктора и якоря имеем следующие уравнения равновесия напряжений в комплексно-векторной форме:

$$\bar{u}_{1k}^u = r_1^u \bar{i}_{1k}^u + \frac{d \bar{\psi}_{1k}^u}{dt}, \quad (3.3)$$

$$0 = r_1^a \bar{i}_{1k}^a + \frac{d \bar{\psi}_{1k}^a}{dt}. \quad (3.4)$$

Мгновенные значения потокоцеплений определяются как суммы проекций векторов потокоцеплений $\bar{\psi}_{1k}^u$ и $\bar{\psi}_{1k}^a$ от токов \bar{i}_{1k}^u и \bar{i}_{1k}^a на магнитную ось рассматриваемой фазы:

$$\psi_{1k}^u = L_s^u i_{1k}^u + \sum_{t=1}^{m_u} M_{uu} i_{1t}^u \cos(\alpha_{1t}^u - \alpha_{1k}^u) + \sum_{t=1}^{m_a} M_{ua} i_{1t}^a \cos(\alpha_{1t}^a + \theta_a - \theta_i - \alpha_{1k}^u), \quad (3.5)$$

$$\psi_{1k}^a = L_s^a i_{1k}^a + \sum_{t=1}^{m_a} M_{aa} i_{1t}^a \cos(\alpha_{1t}^a - \alpha_{1k}^a) + \sum_{t=1}^{m_u} M_{au} i_{1t}^u \cos(\alpha_{1t}^u - \alpha_{1k}^a + \theta_i - \theta_a) \quad (3.6)$$

Электромагнитный процесс в муфте определяется результирующими векторами токов, напряжений и потокоцеплений обмоток индуктора и якоря муфты

$$\begin{aligned} \bar{I}_1^u &= \sum_{t=1}^{m_u} i_{1t}^u e^{j\alpha_{1t}^u}, & \bar{U}_1^u &= \sum_{t=1}^{m_u} u_{1t}^u e^{j\alpha_{1t}^u}, & \bar{\Psi}_{1k}^u &= \sum_{t=1}^{m_u} \psi_{1t}^u e^{j\alpha_{1t}^u}, \\ \bar{I}_1^a &= \sum_{k=1}^{m_a} i_{1k}^a e^{j\alpha_{1k}^a}, & \bar{U}_1^a &= \sum_{k=1}^{m_a} u_{1k}^a e^{j\alpha_{1k}^a}, & \bar{\Psi}_{1k}^a &= \sum_{k=1}^{m_a} \psi_{1k}^a e^{j\alpha_{1k}^a}, \end{aligned} \quad (3.7)$$

Для результирующих векторов токов, напряжений и потокосцеплений имеем следующие уравнения (суммируя уравнения по k):

$$\begin{aligned}\bar{U}_1^u &= r_1^u \bar{I}_1^u + \frac{d\bar{\Psi}_1^u}{dt}, \\ 0 &= r_1^a \bar{I}_1^a + \frac{d\bar{\Psi}_1^a}{dt}\end{aligned}\quad (3.8)$$

В выражениях (3.7) и (3.8) векторы $\bar{U}_1^u, \bar{I}_1^u, \bar{\Psi}_1^u$ определяются в координатной системе d, jq , а векторы $\bar{U}_1^a, \bar{I}_1^a, \bar{\Psi}_1^a$ – в координатной системе d_a, jq_a . Если векторы $\bar{U}_1^u, \bar{I}_1^u, \bar{\Psi}_1^u$ и $\bar{U}_1^a, \bar{I}_1^a, \bar{\Psi}_1^a$ определить в единой системе координат d_0, jq_0 , вращающейся с угловой скоростью Ω_0 , то, как видим из рис. 3.1, в новой системе координат получим равновесия напряжений в общей системе координат d_0, q_0 .

$$\begin{aligned}\bar{I}_{01}^u &= \bar{I}_1^u e^{-j(\theta_0 \pm \theta_i)}, \quad \bar{U}_{01}^u = \bar{U}_1^u e^{-j(\theta_0 \pm \theta_i)}, \quad \bar{\Psi}_{01}^u = \bar{\Psi}_1^u e^{-j(\theta_0 \pm \theta_i)}, \\ \bar{I}_{01}^a &= \bar{I}_1^a e^{-j(\theta_0 - \theta_i)}, \quad \bar{U}_{01}^a = \bar{U}_1^a e^{-j(\theta_0 - \theta_i)}, \quad \bar{\Psi}_{01}^a = \bar{\Psi}_1^a e^{-j(\theta_0 - \theta_i)},\end{aligned}\quad (3.9)$$

Для чего проделаю следующие операции: заменим в выражениях потокосцеплений (3.5) \cos угловполусуммой сопряженных комплексов; умножим правые и левые части уравнений (4) $e^{-j(\theta_0 - \theta_i)}$ и $e^{-j(\theta_0 - \theta_a)}$; при суммировании уравнения (3.4) по k учтем, что угловая частота индуктора Ω_i положительна. В результате получим следующие уравнения:

$$\overline{U}_{01}^u = r_u \overline{I}_{01}^u + \frac{d \overline{\Psi}_{01}^u}{dt} + j(\Omega_0 + \Omega_i) \cdot \overline{\Psi}_{01}^u, \quad (3.10)$$

$$0 = r_a \overline{I}_{1a}^0 + \frac{d \overline{\Psi}_{01}^a}{dt} + j(\Omega_0 - \Omega_a) \cdot \overline{\Psi}_{01}^a$$

$$\overline{I}_{01}^u = \overline{I}_1^u e^{-j(\theta_0 - \theta_i)}$$

$$\overline{U}_{01}^u = \overline{U}_1^u e^{-j(\theta_0 - \theta_i)}, \quad (3.11)$$

$$\overline{\Psi}_{01}^u = \overline{\Psi}_1^u e^{-j(\theta_0 - \theta_i)}$$

$$\overline{I}_{01}^a = \overline{I}_1^a e^{-j(\theta_0 - \theta_a)}$$

$$\overline{\Psi}_{01}^a = \overline{\Psi}_1^a e^{-j(\theta_0 - \theta_a)} \quad (3.12)$$

$$\overline{U}_{01}^a = 0$$

$$\overline{\Psi}_{01}^u = L_s^u \overline{I}_{01}^u + \frac{m_u}{2} [M_{uu} \overline{I}_{01}^u + M_{ua} \overline{I}_{01}^a] \quad (3.13)$$

$$\overline{\Psi}_{01}^a = L_s^a \overline{I}_{01}^a + \frac{m_a}{2} [M_{aa} \overline{I}_{01}^a + M_{au} \overline{I}_{01}^u]$$

Разложив уравнения (3.10) и выражения (3.13) на оси координат \mathbf{d}_o и \mathbf{q}_o , получим обобщённую математическую модель муфты[7]:

$$\begin{aligned}
U_{10\,d}^u &= r_1^u I_{10\,d}^u + \frac{d\Psi_{10\,d}^u}{dt} - (\Omega_0 + \Omega_i) \cdot \Psi_{10\,q}^u \\
U_{10\,q}^u &= r_1^u I_{10\,q}^u + \frac{d\Psi_{10\,q}^u}{dt} + (\Omega_0 + \Omega_i) \cdot \Psi_{10\,d}^u
\end{aligned}
, \tag{3.14}$$

$$0 = r_a I_{10\,d}^a + \frac{d\Psi_{10\,d}^a}{dt} - (\Omega_0 - \Omega_a) \cdot \Psi_{10\,q}^a$$

$$0 = r_a I_{10\,q}^a + \frac{d\Psi_{10\,q}^a}{dt} + (\Omega_0 - \Omega_a) \cdot \Psi_{10\,d}^a$$

$$\Psi_{10\,d}^u = L_s^u I_{10\,d}^u + \frac{m_u}{2} M_{uu} I_{10\,d}^u + \frac{m_u}{2} M_{ua} I_{10\,d}^a = L_u I_{10\,d}^u + L_m I_{10\,d}^a$$

$$\Psi_{10\,q}^u = L_s^u I_{10\,q}^u + \frac{m_u}{2} M_{uu} I_{10\,q}^u + \frac{m_u}{2} M_{ua} I_{10\,q}^a = L_u I_{10\,q}^u + L_m I_{10\,q}^a$$

$$, \tag{3.15}$$

$$\Psi_{10\,d}^a = L_s^a I_{10\,d}^a + \frac{m_a}{2} M_{uu} I_{10\,d}^a + \frac{m_a}{2} M_{ua} I_{10\,d}^u = L_a I_{10\,d}^a + L_m I_{10\,d}^u$$

$$\Psi_{10\,q}^a = L_s^a I_{10\,q}^a + \frac{m_a}{2} M_{uu} I_{10\,q}^a + \frac{m_a}{2} M_{ua} I_{10\,q}^u = L_a I_{10\,q}^a + L_m I_{10\,q}^u$$

$$U_{10\,d}^u = U_{1m}^u \cos((\Omega_1^u - \Omega_0 - \Omega_i)t + \alpha_{u0})$$

$$U_{10\,q}^u = U_{1m}^u \sin((\Omega_1^u - \Omega_0 - \Omega_i)t + \alpha_{u0}), \tag{3.16}$$

где

$U_{10\,d}^u$, $I_{10\,d}^u$, $\Psi_{10\,d}^u$ – проекции векторов напряжения, тока и
потокосцепления индуктора на ось d ;

$U_{10q}^u, I_{10q}^u, \Psi_{10q}^u$ – проекции векторов напряжения, тока и
потокосцепления индуктора на ось q ;

Ω_i – скорость вращения индуктора.

Дифференциальные уравнения (3.14) необходимо дополнить выражением для электромагнитного момента и уравнениями движений индуктора и якоря

$$M_{\vartheta}^a = \Psi_{10d}^u \cdot I_{10q}^u - \Psi_{10q}^u I_{10d}^u \quad (3.16)$$

$$\begin{aligned} -M_{x\partial} - M_{\vartheta}^a + M_{mp}^u &= J_{x\partial}^u \frac{d\Omega_i}{dt} \\ M_{\vartheta}^a - M_g - M_{mp}^{ag} &= J_{ag} \frac{d\Omega_g}{dt}, \quad \Omega_1^u > \Omega_i \quad M_{\vartheta}^a > 0 \quad M_{\vartheta}^u = -M_{\vartheta}^a \end{aligned} \quad (3.17)$$

$$\begin{aligned} -M_{x\partial} + M_{\vartheta}^a + M_{mp}^u &= J_{x\partial}^u \frac{d\Omega_i}{dt} \\ -M_{\vartheta}^a + M_g - M_{mp}^{ag} &= J_{ag} \frac{d\Omega_g}{dt}, \quad \Omega_1^u < \Omega_i \quad M_{\vartheta}^a < 0 \quad M_{\vartheta}^u = M_{\vartheta}^a \end{aligned} \quad (3.18)$$

Полученная система дифференциальных уравнений позволяет рассчитывать неустановившиеся режимы муфты, например методом Рунге-Кутты или разработать виртуальную математическую модель.

Приведенные выше уравнения относятся к двухполюсной муфте. Если муфта многополюсная, то механические данные нужно пересчитать на эквивалентную двухполюсную муфту. Пересчёт производят следующим образом:

– углы поворота индуктора и якоря $\theta_i = p_i \theta_i'$ и $\theta_a = p_a \theta_a'$,

– угловые скорости $\Omega_i = p_i \Omega'_i$ и $\Omega_a = p_a \Omega'_a$,

– вращающийся момент $M_i = M'_i / p_i$ и $M_a = M'_a / p_a$,

причём действительные данные муфты обозначены со штрихом вверху.

Для пересчёта моменты инерции напишем уравнения движений индуктора и якоря, используя действительные данные муфты

$$M'_i - M'_{ci} = J'_i \frac{d\Omega'_i}{dt}$$

$$M'_a - M'_{ca} = J'_a \frac{d\Omega'_a}{dt} \quad (3.19)$$

После подстановки пересчитанных величин величины моментов инерции будут равны

$$J_i = \frac{1}{p_i^2} J'_i J_a = \frac{1}{p_a^2} J'_a \quad (3.20)$$

Для характеристики момента инерции в расчётной практике обычно используют инерционную постоянную H . Если за номинальную скорость вращения якоря принять скорость вращения магнитного поля, созданного токами обмотки индуктора, то получим

$$H_i = \frac{J_i \Omega_i}{M_{ном}} = \frac{J_i \Omega_i^2}{P_{ном}}$$

$$H_a = \frac{J_a \Omega_a}{M_{ном}} = \frac{J_a \Omega_a^2}{P_{ном}} \quad (3.21)$$

Полученная математическая модель позволяет изучать различные режимы реверсивной индукционной муфты скольжения в любой системы координат.

3.3. Выбор системы координат.

Рассмотрим систему уравнений (3.14) для динамического режима. Если выбрать скорость вращения общей системы координат $d_o, j q_o$, равной скорости вращения индуктора Ω_i , рассматриваемая система уравнений будет иметь следующий вид:

$$\begin{aligned} U_{10d}^u &= r_1^u I_{10d}^u + \frac{d\Psi_{10d}^u}{dt} \\ U_{10q}^u &= r_1^u I_{10q}^u + \frac{d\Psi_{10q}^u}{dt} \end{aligned} \quad , (3.22)$$

$$0 = r_a I_{10d}^a + \frac{d\Psi_{10d}^a}{dt} + (\Omega_i + \Omega_a) \cdot \Psi_{10q}^a$$

$$0 = r_a I_{10q}^a + \frac{d\Psi_{10q}^a}{dt} - (\Omega_i + \Omega_a) \cdot \Psi_{10d}^a$$

$$U_{10d}^u = U_{1m}^u \cos(\Omega_u t + \alpha_{u0})$$

$$U_{10q}^u = U_{1m}^u \sin(\Omega_u t + \alpha_{u0}), \quad (3.23)$$

Полученная система дифференциальных уравнений подобна системе дифференциальных уравнений для переходных процессов асинхронного двигателя. Основным отличием системы дифференциальных уравнений для динамических режимов РИМС является наличие скорости вращения индуктора.

ГЛАВА 4. РАЗРАБОТКА ВИРТУАЛЬНОЙ ФУНКЦИОНАЛЬНОЙ МОДЕЛИ СХЕМЫ РЕГУЛИРУЕМОЙ ИНДУКЦИОННОЙ МУФТЫ СКОЛЬЖЕНИЯ

4.1 Пакет визуального программирования Simulink

Одной из наиболее привлекательных особенностей системы MATLAB является наличие в ней наглядного и эффективного средства составления программных моделей — пакета визуального программирования Simulink [8].

Пакет Simulink позволяет осуществлять исследование (моделирование во времени) поведения динамических линейных и нелинейных систем, причем составление «программы» и ввод характеристик систем можно производить в диалоговом режиме, путем сборки на экране схемы соединений элементарных (стандартных или пользовательских) звеньев. В результате такой сборки получается модель системы (называемая S-моделью), которая сохраняется в файле с расширением *.mdl. Такой процесс составления вычислительных программ принято называть визуальным программированием.

S-модель может иметь иерархическую структуру, то есть состоять из моделей более низкого уровня, причем количество уровней иерархии практически не ограничено. В процессе моделирования есть возможность наблюдать за процессами, которые происходят в системе. Для этого используются специальные блоки («обзорные окна»), входящие в состав библиотеки Simulink. Библиотека может быть пополнена пользователем за счет разработки собственных блоков.

Создание моделей в пакете Simulink основывается на использовании технологии Drag-and-Drop (шаг за шагом). В качестве «кирпичиков» при

построении S-модели применяются визуальные блоки (модули), которые сохраняются в библиотеках Simulink.

Библиотека блоков Simulink (рисунок 4.1) – это набор визуальных объектов, при использовании которых, соединяя отдельные блоки между собой линиями связей, можно составлять функциональную блок-схему любого устройства.

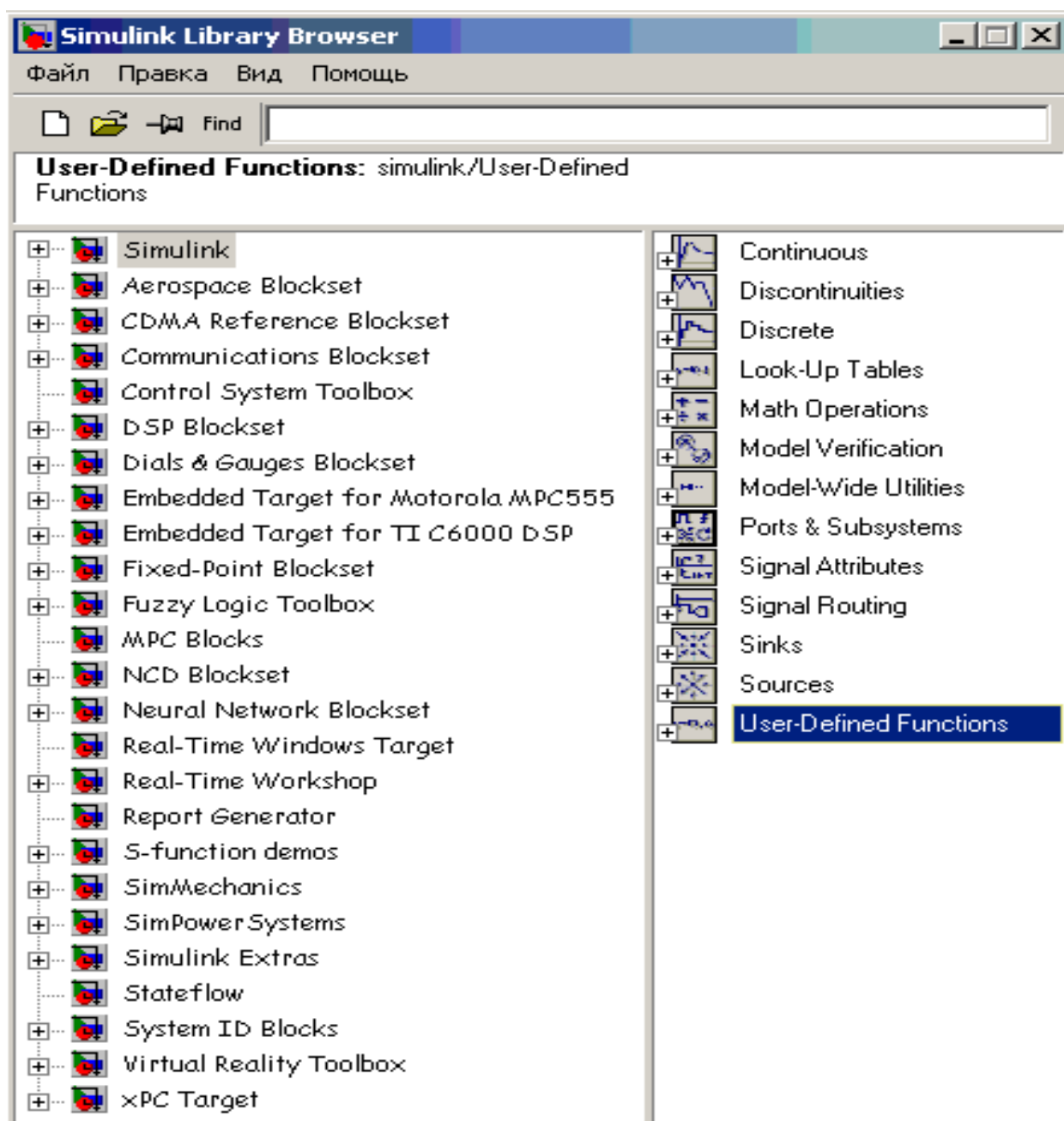


Рисунок 4.1 - Окно SimulinkLibraryBrowser

Сборка блок-схемы S-модели заключается в том, что графические изображения выбранных блоков с помощью мыши перетягиваются из окна раздела библиотеки в окно блок-схемы, а затем выходы одних блоков в окне блок-схемы соединяются со входами других блоков (также с помощью мыши). Соединение блоков выполняется следующим образом: указатель мыши подводят к определенному выходу нужного блока (при этом указатель должен приобрести форму крестика), нажимают левую кнопку и, не отпуская ее, перемещают указатель к нужному входу другого блока, а потом отпускают кнопку. Если соединение осуществлено верно, на входе последнего блока появится изображение черной стрелки. Сборка модели осуществляется в рабочем поле специального окна (рисунок 4.2). Это окно имеет строку меню, панель инструментов и рабочее поле. Меню File (Файл) содержит команды, предназначенные для работы с MDL - файлами; меню Edit (Правка) — команды редактирования блок-схемы; меню View (Вид) команды изменения внешнего вида окна; меню Simulation (Моделирование) — команды управления процессом моделирования; меню Format (Формат) — команды редактирования формата (то есть команды, позволяющие изменить внешний вид отдельных блоков и блок-схемы в целом). Меню Tools (Инструменты) включает некоторые дополнительные сервисные средства, предназначенные для работы с S-моделью.

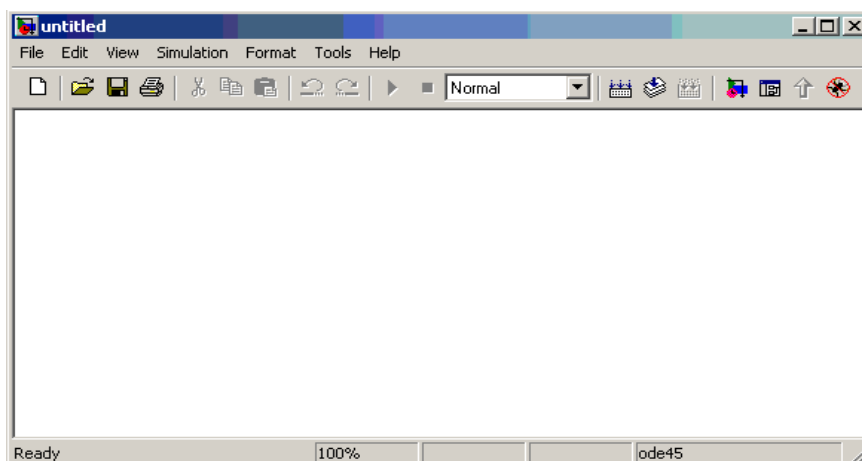


Рисунок 4.2 - Окно, в котором осуществляется сборка модели.

Любая блок-схема моделируемой системы должна включать в себя один или несколько блоков-источников, генерирующих сигналы, которые, собственно, и вызывают «движение» моделируемой системы, и один или несколько блоков-приемников, которые позволяют получить информацию о выходных сигналах этой системы (увидеть результаты моделирования). Запуск модели на выполнение осуществляется нажатием на кнопку ►, либо через меню Simulation→Start, остановка нажатием на кнопку ■, либо через меню Simulation→Stop, пауза - на кнопку ||, либо через меню Simulation→Pause, пауза активна, когда модель запущена на выполнение. Кнопки расположены на панели инструментов.

4.2. Параметры модели для РИМС серии 4А

Для моделирования параметры РИМС были взяты параметры асинхронного двигателя с короткозамкнутым ротором марки 4А112М4У3:

- номинальная выходная мощность $P_{2н}=5.5$ кВт,
- номинальное фазное напряжение обмотки статора $U_{1н}=220$ В,
- номинальная частота тока $f_1=50$ Гц,
- номинальный коэффициент полезного действия $\eta_n = 85.5$ %,
- номинальный коэффициент мощности статорной обмотки $\cos\varphi=0.85$,
- критическое скольжение ротора $S_k= 25$ %,
- номинальное скольжение ротора $S_n= 3.6$ %,
- число пар полюсов: $p=2$,
- число фаз: $m=3$,
- скорость холостого хода: $n_1=1500$ об/мин,
- момент инерции на валу машины: $J=0,017$ кг \times м²,
- параметры Г-образной схемы замещения в режиме короткого замыкания (рисунок 4.3) в относительных единицах:
- в номинальном режиме:

$R_1^*=0.064$, $X_1^*=0.078$, $R_2^{**}=0.041$, $X_2^{**}=0.13$, $X_m^*=2.8$,

- в режиме короткого замыкания:

$R_2^{**k3}=0.048$, $X_2^{**}=0.062$.

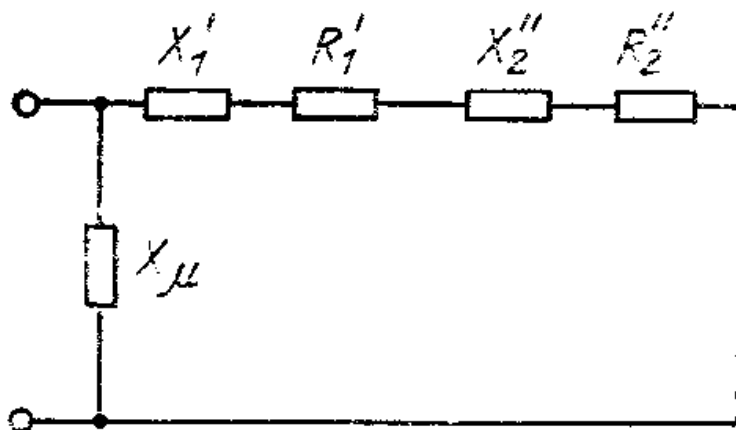


Рисунок 4.3 – Г-образная схема замещения

По известным паспортным данным РИМС и параметрам Г-образной схемы замещения рассчитываются параметры Т-образной схемы замещения в режиме короткого замыкания (рисунок 4.4) и коэффициенты системы уравнений и параметры блоков модели РИМС.

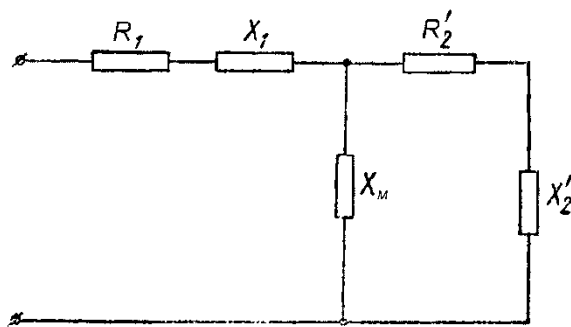


Рисунок 4.4 – Т-образная схема замещения

4.3. Обобщенная структурная модель регулируемой индукционной муфты скольжения

Регулируемую индукционную муфту скольжения (РИМС) можно представить в виде электромеханического многополюсника (рис.4.5), имеющего три пары электрических зажимов, соответствующих обмоткам муфты, к которым подведены напряжения u_1, u_2, u_3 , и две пары механических зажимов, представляющих соответственно безынерционные якорь и индуктор муфты, на которых при скоростях Ω_a и Ω_u в результате электромеханического преобразования энергии развивается электромагнитный момент M_ε , действующий и на якорь и на индуктор муфты.

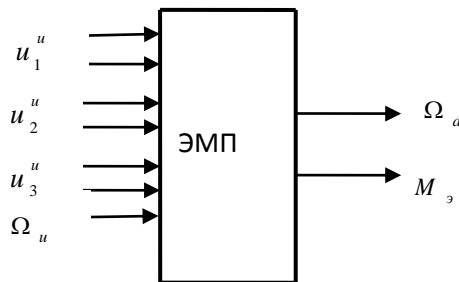


Рис.4.5. Электромеханический многополюсник

Переменные $M_\varepsilon, \Omega_a, \Omega_u$ связывают электромеханический преобразователь с механической частью системы, а напряжения u_1, u_2, u_3 — с системой управления электроприводом.

Для построения математического описания динамических процессов в регулируемой индукционной муфте скольжения [6], будем использовать её математическую модель, которая имеет следующий вид:

$$U_d^u = \pm r^u I_d^u \pm \frac{d\Psi_d^u}{dt} \mp (\Omega_0 - \Omega^u) \cdot \Psi_q^u$$

$$\begin{aligned}
U_q^u &= \pm r^u I_q^u \pm \frac{d\Psi_q^u}{dt} \pm (\Omega_0 - \Omega^u) \cdot \Psi_d^u \\
0 &= r^a I_d^a + \frac{d\Psi_d^a}{dt} - (\Omega_0 - \Omega_{\text{эл}}^a) \cdot \Psi_q^a \\
0 &= r^a I_q^a + \frac{d\Psi_q^a}{dt} + (\Omega_0 - \Omega_{\text{эл}}^a) \cdot \Psi_d^a
\end{aligned} \tag{4.1}$$

$$\begin{aligned}
\mp M_{n\partial} \pm M_{\text{я}}^u &= J_{n\partial}^u \frac{d\Omega^u}{dt}, \\
\pm (M_{\text{я}}^a - M_c^a) &= J_a \frac{d\Omega^a}{dt}, \quad \Omega_{\text{эл}}^a = p\Omega^a,
\end{aligned} \tag{4.2}$$

$$M_{\text{я}}^a = \Psi_d^u \cdot I_q^u - \Psi_q^u I_d^u M_{\text{я}}^u = -M_{\text{я}}^a \tag{4.3}$$

$$\Psi_d^u = L^u I_d^u + \frac{3}{2} M^{ua} I_d^a \Psi_q^u = L^u I_q^u + \frac{3}{2} M^{ua} I_q^a \tag{4.4}$$

$$\Psi_d^a = L^a \cdot I_d^a + M^{au} I_1^u \Psi_q^a = L^a I_q^a + M^{au} I_q^u \tag{4.5}$$

где $U_d^u, I_d^u, \Psi_d^u, U_q^u, I_q^u, \Psi_q^u$ – напряжения, токи и потокосцепления индуктора по осям ***d*** и ***jq***;

$I_d^u, \Psi_d^u, I_q^u, \Psi_q^u$ – токи и потокосцепления якоря по осям ***d*** и ***jq***;

Очевидно, что для построения структурной модели РИМС необходимо выбрать значение скорости вращения системы координат Ω_0 , из уравнений (4.1) с помощью выражений (4.4, 4.5) исключить зависимые переменные и выбрать необходимое уравнение вычисления электромагнитного момента. Таким образом, в общем виде структурная модель муфты на функциональном уровне может быть представлена с

помощью пяти функциональных блоков, схема соединения которых приведена на рис. 4.6.

Функциональный блок **ЕМР** имеет четыре входа: U_d^u , U_q^u , $\Omega_{эл}^a$, Ω^u , так как ротор короткозамкнутый. На основании вводимых напряжений индуктора U_d^u , U_q^u , скоростей индуктора Ω^u , якоря $\Omega_{эл}^a$ осуществляется вычисление мгновенных значений четырех составляющих вектора $\overline{\Psi}$.

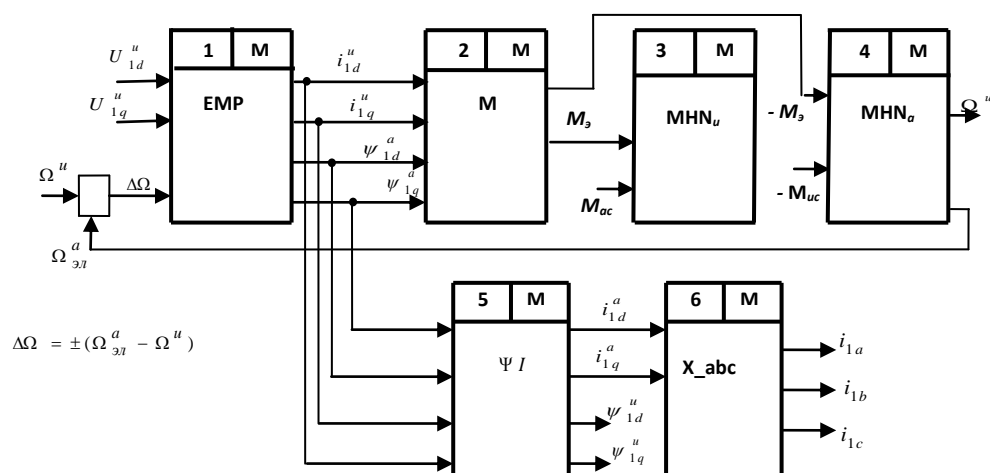


Рис. 4.6. Схема обобщенной структурной модели РИСМ.

Функциональный блок **М** предназначен для вычисления мгновенных значений электромагнитного момента m_s^a и m_s^u . Внутреннее представление этого блока определяется выбранным вариантом компонент вектора потокоцеплений $\overline{\Psi}$.

Внутреннее представление функционального блока **МНН** не зависит от системы координат и рассчитывает скорость вращения якоря.

Блок **X_ALB**, который соответственно формирует трехфазную систему напряжений u_1^u , u_2^u , u_3^u

Разработанная структурная модель РИСМ позволяет исследовать динамические режимы.

Автономные системы электроснабжения на самолётах, - это системы в которых для привода генераторов используются ходовые авиадвигатели. Основной недостаток ходовых двигателей – это изменяющаяся частота вращения выходного вала в зависимости от режима его работы. Получение постоянной частоты переменного тока генератора при изменяющейся частоте вращения ходового двигателя, сбросе и набросе нагрузки на синхронный генератор и других динамических режимах исследования их является важной задачей. Одним из приводов постоянных оборотов является привод на базе реверсивной индукционной муфты скольжения (РИМС)[9].

Авторами предлагается разработанная схема функциональных блоков расчёта токов и потокосцеплений, момента и угловой скорости якоря, позволяющая исследовать переходные процессы в РИМС при независимой угловой частоте индуктора. Наибольший практический интерес представляют структурные модели РИМС, построенные в ортогональной системе координат d, jq , вращающейся с частотой, равной угловой частоте индуктора $\Omega_0 = \Omega_i$. Это связано с тем, что получаемые модели могут быть использованы для имитационного моделирования систем, в которых РИМС подключается к источникам несинусоидального напряжения (например, статические преобразователи частоты).

$$U_{1d}^u = U_{1m}^u \cos(\omega_1^u t + \alpha_{10}^u); U_{1q}^u = U_{1m}^u \sin(\omega_1^u t + \alpha_{10}^u) \quad (4.6)$$

$$U_{o1d}^u = r_u I_{o1d}^u + \frac{d}{dt} \Psi_{o1d}^u; U_{o1q}^u = r_u I_{o1q}^u + \frac{d}{dt} \Psi_{o1q}^u$$

$$0 = r_a I_{o1d}^a + \frac{d}{dt} \Psi_{o1d}^a + (\Omega_i + \Omega_a) \Psi_{o1q}^a$$

$$0 = r_a I_{o1q}^a + \frac{d}{dt} \Psi_{o1q}^a - (\Omega_i + \Omega_a) \Psi_{o1d}^a \quad (4.7)$$

$$M_e = \frac{m_u}{3} (\Psi_{o1d}^a \cdot I_{o1q}^u - \Psi_{o1q}^a \cdot I_{o1d}^u); \frac{d}{dt} \Omega_a = \frac{M_e - M_c}{J_a}, \quad (4.8)$$

где в системе координат d, jq даны: выражения для напряжений (4.6); уравнения динамического режима (4.7); выражения электромагнитного момента и уравнения движения якоря (4.8).

Четыре уравнения системы (4.6) содержат восемь линейно зависимых

переменных. В этой связи, здесь, прежде всего необходимо, с помощью выражений связи токов и потокоцеплений обмоток индуктора и якоря исключить две пары переменных, то есть выбрать состав компонент вектора $[\bar{\Psi}]$

$$\Psi_{o1d}^u = L_u I_{o1d}^u + M_{ua} I_{o1d}^a; \Psi_{o1q}^u = L_u I_{o1q}^u + M_{ua} I_{o1q}^a$$

$$\Psi_{o1d}^a = L_a I_{o1d}^a + M_{ua} I_{o1d}^u; \Psi_{o1q}^a = L_a I_{o1q}^a + M_{ua} I_{o1q}^u \quad (4.9)$$

Выберем состав компонент вектора $\bar{\Psi}$ так, чтобы результирующее математическое описание электромеханического преобразования энергии в РИМС не содержало бы модель безинерционных контуров.

Для этого из двух последних уравнений системы (4.9) определяем токи якоря

$$I_{o1d}^a = \frac{1}{L_a} \Psi_{o1d}^a - \frac{M_{ua}}{L_a} I_{o1d}^u;$$

$$I_{o1q}^a = \frac{1}{L_a} \Psi_{o1q}^a - \frac{M_{ua}}{L_a} I_{o1q}^u \quad (4.10)$$

Полученные выражения подставим в уравнения для потокосцеплений индуктора, то есть в первые два уравнения системы

$$\Psi_{o1d}^u = \left(L_u - \frac{(M_{ua})^2}{L_a} \right) I_{o1d}^u + \frac{M_{ua}}{L_a} \Psi_{o1d}^a$$

$$\Psi_{o1q}^u = \left(L_u - \frac{(M_{ua})^2}{L_a} \right) I_{o1q}^u + \frac{M_{ua}}{L_a} \Psi_{o1q}^a \quad (4.11)$$

Обозначив $L_\sigma^u = L_u - \frac{(M_{ua})^2}{L_a}$, $s = d/dt$ и проделав ряд преобразований, как показано в[2], получим систему уравнений вида, удобного для построения функционального блока расчета токов и потокосцеплений регулируемой индукционной муфты скольжения (рис.4.7)

$$I_{o1q}^u = \frac{1}{r_u + sL_\sigma^u} \left(U_{o1q}^u - \frac{M_{ua}}{L_a} s \Psi_{o1q}^a \right)$$

$$\Psi_{o1d}^a = \frac{1}{s} \left(\frac{R_a M_{ua}}{L_a} I_{o1d}^u - \frac{R_a}{L_a} \Psi_{o1d}^a - (\Omega_i + \Omega_a) \Psi_{o1q}^a \right)$$

$$I_{o1d}^u = \frac{1}{R_u + sL_\sigma^u} \left(U_{o1d}^u - \frac{M_{ua}}{L_a} s \Psi_{o1d}^a \right)$$

$$\Psi_{o1q}^a = \frac{1}{s} \left(\frac{R_a M_{ua}}{L_a} I_{o1q}^u - \frac{R_a}{L_a} \Psi_{o1q}^a + (\Omega_i + \Omega_a) \Psi_{o1d}^a \right)$$

$$M_e = a_{15}(I_{o1q}^u \Psi_{o1d}^a - I_{o1d}^u \Psi_{o1q}^a); a_{15} = \left(\frac{m_{up}}{2}\right) \left(\frac{M_{ua}}{L_a}\right)$$

$$U_{1d}^u = U_{1m}^u \cos(\omega_1^u t + \alpha_{10}^u)$$

$$U_{1q}^u = U_{1m}^u \sin(\omega_1^u t + \alpha_{10}^u)$$

Используя программу MATLAB[8] и вышеуказанные уравнения, были разработаны функциональные блоки расчёта электромагнитного момента, токов и потокосцеплений и угловой частоты якоря РИМС.

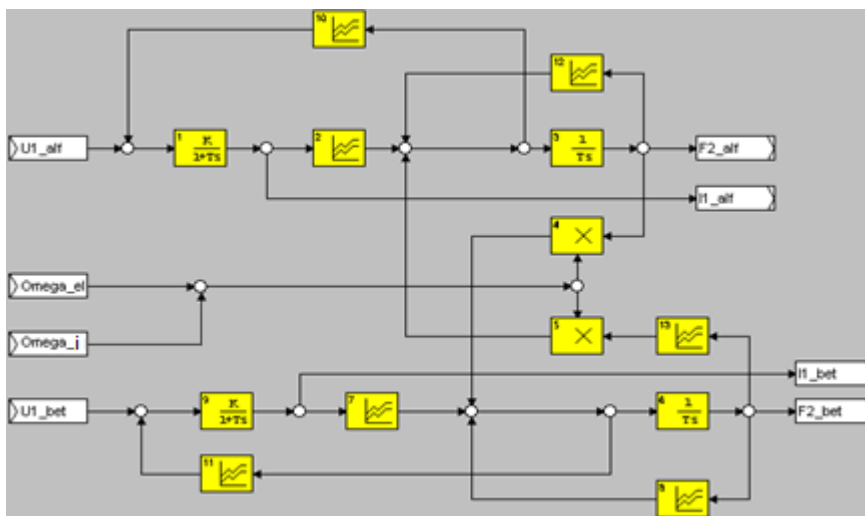


Рис.4.7. Функциональный блок расчёта токов и потокосцеплений.

На рис.4.7 приведен функциональный блок расчёта токов и потокосцеплений, в котором введен элемент получения электрической угловой частоты с учетом угловой частоты индуктора, что является особенностью данного функционального блока (Таблица 1).

Таблица 1.

Значения параметров функционального блока расчета токов и потокосцеплений регулируемой индукционной муфты скольжения

№ элемента	Значения параметров	
	Имя	Параметр
1	K_1	$\frac{1/R_u}{1 + sT_1}$
	T_1	$\frac{L_\sigma^u}{R_u}$
2	K_2	$\frac{R_a M_{ua}}{L_a}$
3	T	1
6	T	1
7	K_2	$\frac{R_a M_{ua}}{L_a}$
8	K_3	$-\frac{R_a}{L_a}$
9	K_1	$\frac{L_\sigma^u}{R_u}$
	T_1	$\frac{R_a M_{ua}}{L_a}$
10	K_1	1
11	K_1	1
12	K_3	$\frac{R_a M_{ua}}{L_a}$
13	K	-1

Таким образом, разработанная схема функциональных блоков позволяет проводить исследование переходных процессов РИМС.

4.4. Разработка виртуальной функциональной модели регулируемой индукционной муфты скольжения

Выберем состав компонентов вектора потокосцепления $\bar{\Psi}$ так, чтобы результирующее математическое описание электромеханического преобразования энергии в РИМС не содержало бы модель безинерционных контуров.

Наибольший практический интерес представляют структурная модель РИМС построенная во вращающейся ортогональной системе координат d, jq с угловой скоростью индуктора Ω_i .

$$\begin{aligned} U_{o1d}^u &= r_u I_{o1d}^u + \frac{d}{dt} \Psi_{o1d}^u; & U_{o1q}^u &= r_u I_{o1q}^u + \frac{d}{dt} \Psi_{o1q}^u; \\ U_d^a &= r^a I_d^a + \frac{d}{dt} \Psi_d^a + (\Omega_i + \Omega_a) \Psi_q^a; & U_q^a &= r^a I_q^a + \frac{d}{dt} \Psi_q^a - (\Omega_i + \Omega_a) \Psi_d^a; \end{aligned}$$

Вышеуказанная система содержит восемь линейно зависимых переменных. В этой связи, здесь, прежде всего необходимо, с помощью выражений связи токов и потокосцеплений обмоток статора и ротора исключить две пары переменных, то есть выбрать состав компонент вектора потокосцепления $\bar{\Psi}$. Выбор состава компонентов осуществлен так, чтобы в результирующем математическом описании электрического преобразования энергии в РИМС отсутствуют инерционные контуры. Прделав операции как в [3], получим выражения

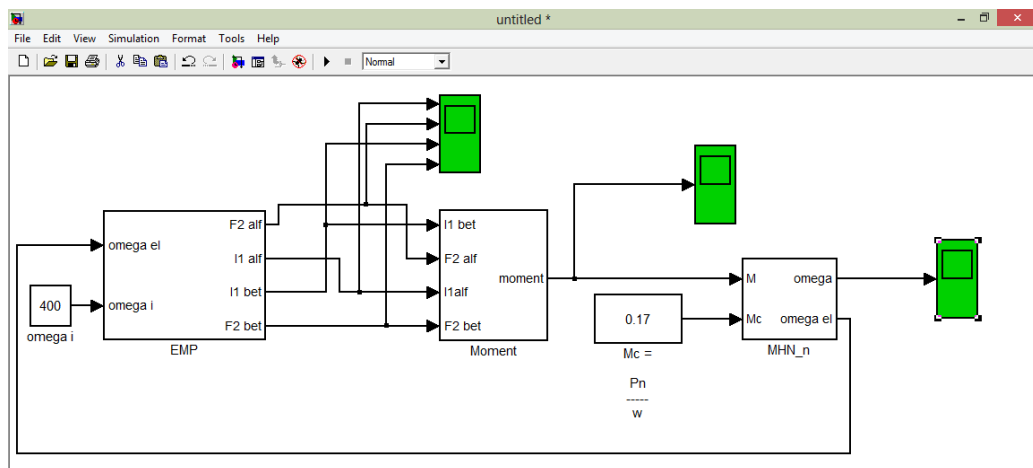
$$\begin{aligned} I_{o1d}^u &= \frac{1}{R_u + sL_\sigma^u} \left(U_{o1d}^u - \frac{M_{ua}}{L_a} s \Psi_{o1d}^a \right) & I_{o1q}^u &= \frac{1}{r_u + sL_\sigma^u} \left(U_{o1q}^u - \frac{M_{ua}}{L_a} s \Psi_{o1q}^a \right) \\ \Psi_{o1d}^a &= \frac{1}{s} \left(\frac{R_a M_{ua}}{L_a} I_{o1d}^u - \frac{R_a}{L_a} \Psi_{o1d}^a - (\Omega_i + \Omega_a) \Psi_{o1q}^a \right) \end{aligned}$$

$$\Psi_{o1q}^a = \frac{1}{s} \left(\frac{R_a M_{ua}}{L_a} I_{o1q}^u - \frac{R_a}{L_a} \Psi_{o1q}^a + (\Omega_i + \Omega_a) \Psi_{o1d}^a \right)$$

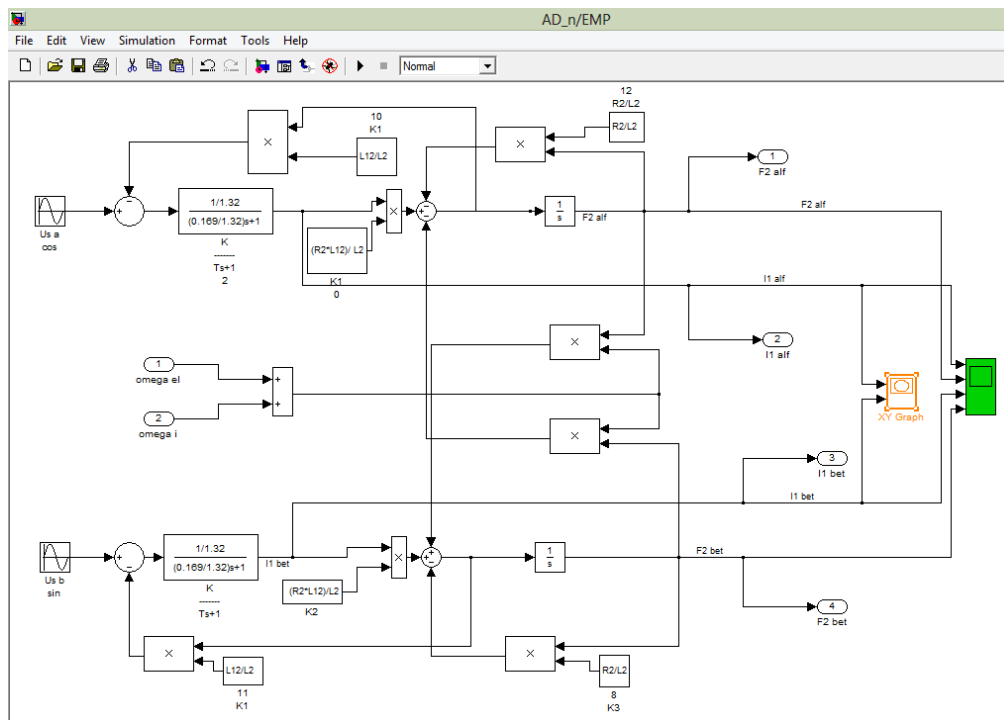
$$M_e = a_{15} (I_{o1q}^u \Psi_{o1d}^a - I_{o1d}^u \Psi_{o1q}^a); \quad a_{15} = \left(\frac{m_u p}{2} \right) \left(\frac{M_{ua}}{L_a} \right)$$

$$U_{1d}^u = U_{1m}^u \cos(\omega_1^u t + \alpha_{10}^u) \quad U_{1q}^u = U_{1m}^u \sin(\omega_1^u t + \alpha_{10}^u)$$

Используя программу MATLAB [3] и вышеуказанные уравнения, были разработаны функциональные блоки расчёта электромагнитного момента, токов и потокосцеплений и угловой частоты якоря РИМС.



a)



б)

Рис. 4.8

а - структурная модель;

б - функциональный блок расчёта токов и потокосцеплений.

Блоки Us_α и Us_β (рисунок 4.9) являются генераторами гармонических сигналов, Us_α – косинусоиды, Us_β – синусоиды. Они имитируют работу источников напряжения.

Настраиваемыми параметрами являются:

Sinetype – тип синусоидальной волны,

Amplitude - амплитуда сигнала, для данной схемы $\sqrt{2} \cdot U_{1n} = \sqrt{2} \cdot 220 = 311 \text{ В}$,

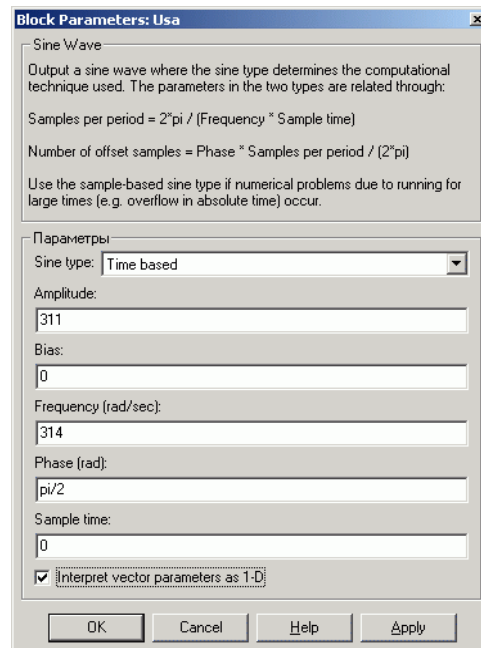
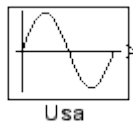
Bias – смещение (постоянная составляющая синусоиды),

Frequency – угловая частота колебаний, для данной схемы равная ω ,?

Phase – начальная фаза (в радианах), равная:

$\frac{\pi}{2}$ для косинусоиды, 0 – для синусоиды,

Sampletime – величина дискрета времени.



а)

б)

Рисунок 4.10 - Блок Usa: а) внешний вид, б) окно параметров

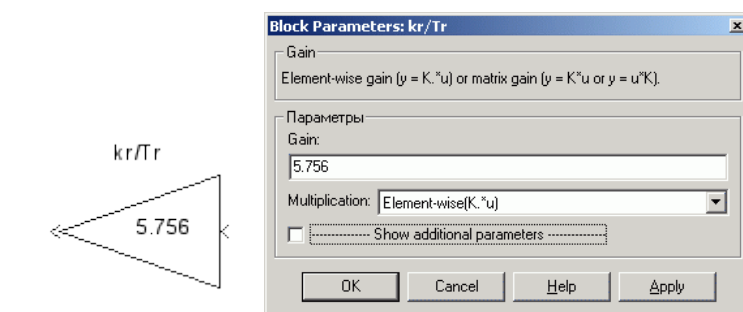
Блок k_R / T_R (рисунок 4.11) осуществляет умножение входного сигнала на постоянную величину, значение которой задаётся в настройке блока.

Аналогичные в схеме блоки: $k_R \cdot p$, $k_R \cdot R_R$, p , $k_R \cdot 1.5 \cdot p$, $1/J$, F .

Настраиваемыми параметрами являются:

Gain – коэффициент усиления, для данной схемы $k_R / T_R = 5.756$,

Multiplication – тип способа умножения.



а)

б)

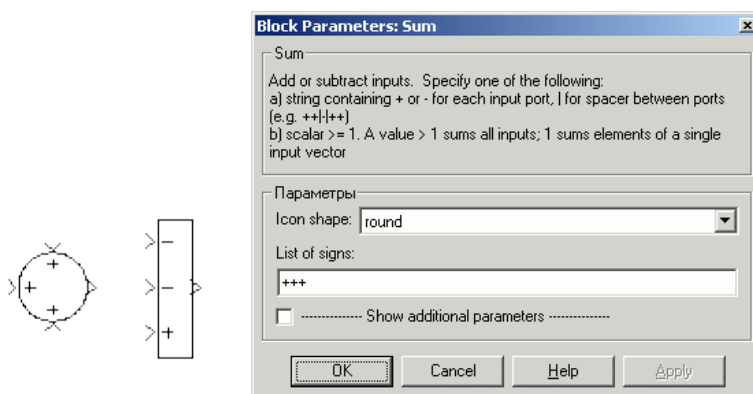
Рисунок 4.11 - Блок k_r / T_r : а) внешний вид, б) окно параметров

Блок Sum (рисунок 4.12) суммирует поступающие на него сигналы (в том числе с разными знаками).

Настраиваемыми параметрами являются:

Iconshape – форма изображения (круг или прямоугольник),

Listofsigns – список входов и их знаки.



а)

б)

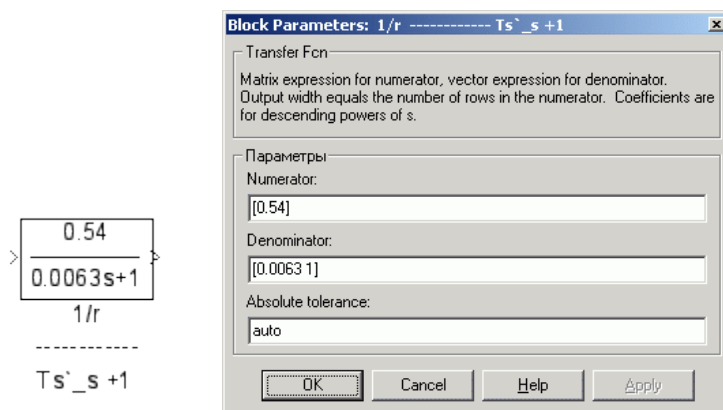
Рисунок 4.12 - Блок Sum: а) внешний вид, б) окно параметров

Блок $\frac{1/r}{T_s s + 1}$ (рисунок 4.13) реализует звено введенной в него передаточной функции. Аналогичный в схеме блок: $\frac{T_R}{T_R s + 1}$.

Настраиваемыми параметрами являются:

Numerator – числитель, для данной схемы $1/r = 0.54$,

Denominator – делитель, для данной схемы $T_s = 0.0063$.



а)

б)

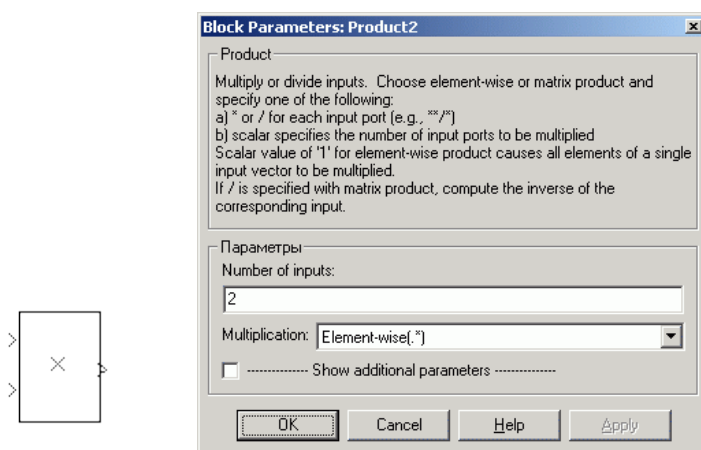
Рисунок 4.13 - Блок $\frac{1/r}{T_s s + 1}$: а) внешний вид, б) окно параметров

Блок Product (рисунок 4.14) выполняет умножение (деление) входных сигналов.

Настраиваемыми параметрами являются:

Numberofinputs – количество входов,

Multiplication – тип способа умножения.



а)

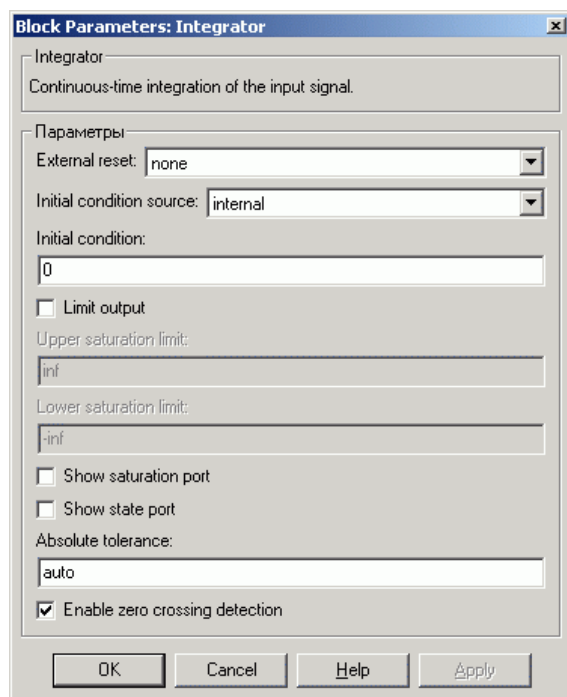
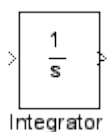
б)

Рисунок 4.14 - Блок Product: а) внешний вид, б) окно параметров

Блок Integrator (рисунок 4.15) представляет идеальное интегрирующее звено. Он позволяет осуществить интегрирование поступающего на него сигнала в непрерывном времени.

Настраиваемыми параметрами являются:


- Externalreset – подключение дополнительного управляющего сигнала,
- Initialconditionsource – определение источника (внутренний или внешний),
- Initialcondition – начальное значение выходной величины,
- Limitoutput – ограничение величины выхода,
- Uppersaturationlimit – верхнее предельное значение выходной величины,
- Lowersaturationlimit – нижнее предельное значение выходной величины,
- Showsaturationport – показать порт насыщения,
- Showstateport – показать порт состояния,
- Absolutetolerance – допустимая предельная величина абсолютной погрешности.



а)

б)

Рисунок 4.15 - Блок $\alpha \cdot M2$: а) внешний вид, б) параметры

Графический дисплей «wm, $M=f(t)$ » (рисунок 4.16) позволяет в ходе моделирования наблюдать графики переходных процессов скорости и момента во времени. По горизонтальной оси откладывается значение модельного времени, а по вертикали значение входной величины, отвечающее этому моменту времени. Окно параметров вызывается нажатием на иконку .

Настраиваемыми параметрами являются:

Numberofaxes – количество осей,

Timerange – интервал времени,

Ticklabels – метки осей,

Sampling – используется только для дискретных во времени процессов. Его значение (1), установленное по умолчанию, для непрерывных процессов изменять не рекомендуется. Позволяет задать периодичность (через сколько дискретов времени) отображения значений времени.

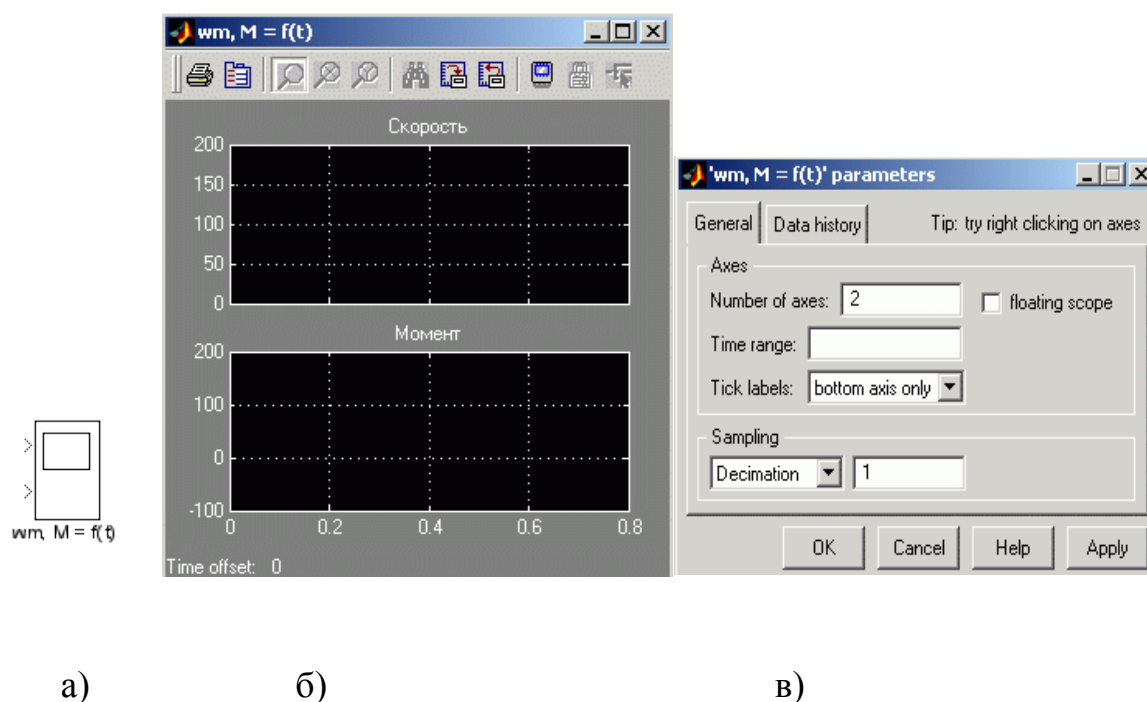


Рисунок 4.16 - Блок «wm, M=f(t)»:

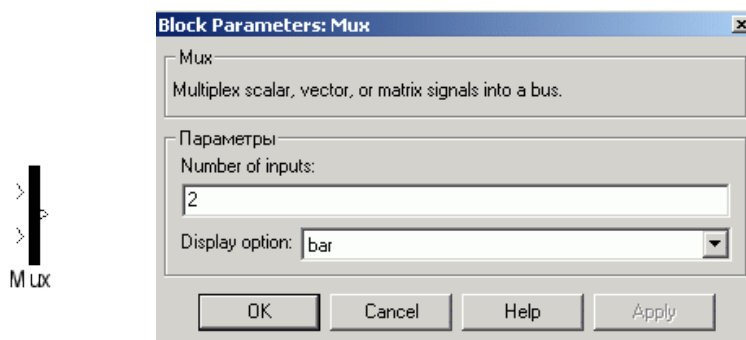
а) внешний вид, б) внутренний вид, в) окно параметров

Шинный формирователь Mux (рисунок 4.17) выполняет объединение входных величин в единый выходной вектор (шину), что очень удобно, так как схема получается мене загромождённой.

Настраиваемыми параметрами являются:

Numberofinputs – число входов,

Displayoption – вид отображения блока.



а)

б)

Рисунок 4.17 - Блок Mux: а) внешний вид, б) окно параметров

Цифровой дисплей «wm, M» (рисунок 4.18) выводит на экран числовые значения входящих в блок величин (скорости и момента).

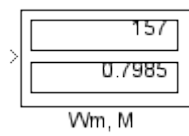
Настраиваемыми параметрами являются:

Format – формат вывода чисел,

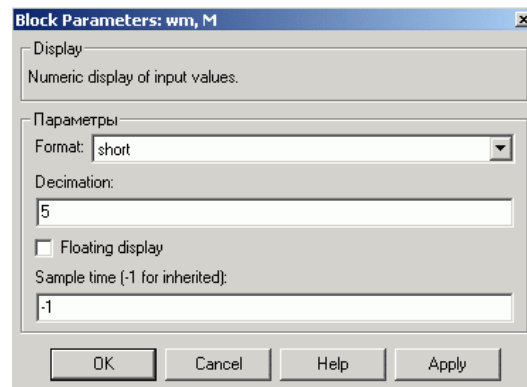
Decimation – позволяет задать периодичность (через сколько дискретов времени) отображения значений времени,

Sampletime – используется только для дискретных во времени процессов.

Его значение (-1), установленное по умолчанию, для непрерывных процессов изменять не рекомендуется.



а)



б)

Рисунок 4.18 - Блок «wm, M»: а) внешний вид, б) окно параметров

4.5 Результаты моделирования

После запуска схемы модели на моделирование и завершения процедуры моделирования можно проанализировать полученные результаты.

Графический дисплей «wm, $M=f(t)$ » отображает переходной процесс скорости и момента во времени, представленный на рисунке 4.19.

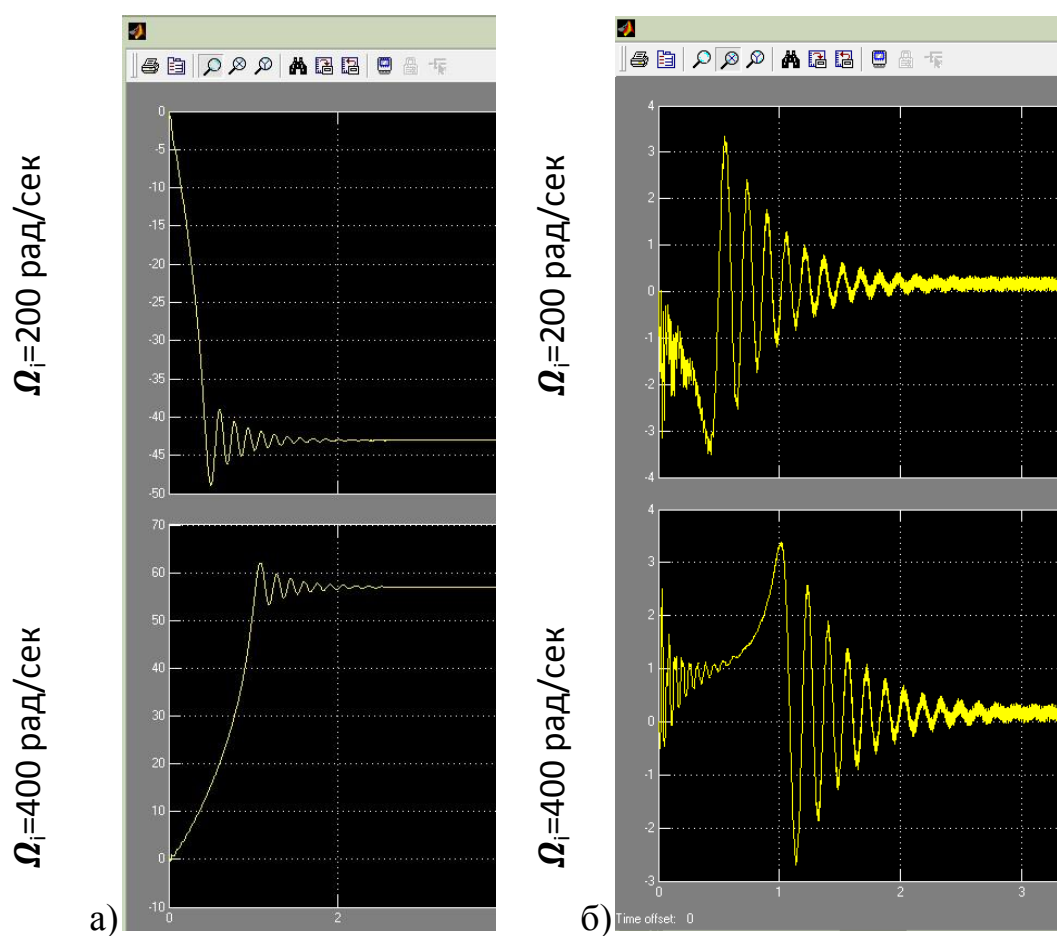


Рис.4.19. а – график скорости якоря; б – график момента на якоре.

Из рисунка 4.19 видно, что при прямом пуске вначале наблюдаются значительные колебания момента и скорости. При приложении момента

нагрузки, аналогично наблюдаются колебания момента и скорости, но менее значительные, чем при пуске, также видно, что при приложении момента нагрузки наблюдается уменьшение скорости.

ЗАКЛЮЧЕНИЕ

Были изучены принцип работы и конструкция РИМС, основные режимы работы.

На основе дифференциальных уравнений переходного режима РИМС была разработана виртуальная функциональная модель РИМС в программе МАТЛАБ. Для проверки правильности полученной модели РИМС была разработана виртуальная лабораторная схема для исследования динамических режимов.

Исследование на виртуальной лабораторной схемы модели РИМС подтвердила правильность полученной виртуальной модели РИМС.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. А.Е. Загорский, Ю.Г. Шакарян, Управление переходными процессами в электрических машинах переменного тока, М., 1986 г.
2. Герман - Галкин С.Г. Компьютерное моделирование полупроводниковых систем в MATLAB 6.0: Учебное пособие – СПб.: Корона принт, 2001., ил.
3. Дадажанов А., Алиев А.Ш., Регулированная индукционная муфта скольжения, Патент на изобретение, IDP 2455, зарегистрированный в Государственном реестре изобретений РУз от 23.11.94 г.
4. Ш.И. Лутидзе, Основы теории электрических машин с управляемым полупроводниковым коммутатором, М., 1968 г.
5. А.Ш. Алиев, Математическая модель реверсивной индукционной муфты скольжения, Республиканская научно-техническая и производственная конференция «Проблемы развития авиакосмической отрасли Республики Узбекистан», 2007 г.
6. К.П. Ковач, И. Рац, Переходные процессы в машинах переменного тока, М – Л, 1963 г.
7. Колганов Алексей Руфимович, Моделирование электромеханических систем, Иван. гос. энерг. ун-т. - Иваново, 2001. Часть 2.
8. Черных И.В., Моделирование электротехнических устройств в MATLAB - М.: ДМК Пресс, 2007.
9. И.М. Синдеев, А.А. Савелов. Системы электроснабжения воздушных судов - М.: Транспорт, 1990.
10. Черных И. В. SIMULINK: Среда создания инженерных приложений / Под общ. ред. В. Г. Потемкина. – М.: ДИАЛОГ МИФИ, 2003.

11. Герман Галкина С. Г., Кардонов Г. А. Электрические машины. Лабораторные работы на ПК. – СПб.: КОРОНА принт, 2003.
12. Герман Галкин С. Г. Компьютерное моделирование полупроводниковых систем в MATLAB 6.0: Учебное пособие. – СПб.: КОРОНА принт, 2001.
13. Герман Галкин С. Г. Линейные электрические цепи. Лабораторные работы. – СПб.: Учитель и ученик, КОРОНА принт, 2002.
14. Герман Галкин С. Г. Силовая электроника. Лабораторные работы. – СПб.: Учитель и ученик, КОРОНА принт, 2002.
15. Дьяконов В. П. MATLAB 6/6.1/6.5 + Simulink 4/5. Основы применения. Полное руководство пользователя. – М.: Солон Пресс. – 2002.
16. Дьяконов В. П. Simulink 4. Специальный справочник. – СПб.: Питер, 2002.– 528 с.
17. Разевиг В. Д. Система проектирования OrCAD 9.2 / В. Д. Разевиг. – М.: Солон Р, 2001.
18. Dommel H. Digital Computer Solution of Electromagnetic Transients in Single and Multiple Networks, *IEEE Transactions on Power Apparatus and Systems*, Vol. PAS 88, No. 4, April, 1969.
19. Recommended Practice for Excitation System Models for Power System Stability Studies, *IEEE Standard 421.5_1992*, August, 1992.
20. SimPowerSystems 4. Reference. The MathWorks, Inc. 2006.
21. Компьютерные технологии в электроприводе: конспект лекций сост. А. В. Доманов. – Ульяновск : УлГТУ, 2006.
22. Лазарев Ю. Моделирование процессов и систем в MATLAB. Учебный курс. - СПб.: Питер; Киев: Издательская группа BHV, 2005. 512 с.: ил.
23. Китаев В. Е., Шляпинтох Л. С. Электротехника с основами промышленной электроники. Учебное пособие для проф.-техн. учебных заведений. Изд. 2-е, переработ. и доп. М., «Высш. школа», 1968.

24. Яковлев Г.С., Магаршак Б.Г., Маникин А. И. Судовые электрические машины. Л., «Судостроение», 1972.

25. Справочник по электрическим машинам: В 2 т./С74. Под общ. ред. Копылова И.П. и Клонова Б.К. Т.1.-М.: Энергоатомиздат, 1988. , ил.

Интернет ресурсы

- 26. http://www.nsu.ru/matlab/MatLab_RU/matlab/default.asp.htm
http://www.nsu.ru/matlab/MatLab_RU/simulink/book2/default.asp.htm
- 27. http://www.nsu.ru/matlab/MatLab_RU/simulink/default.asp.htm
- 28. http://www.nsu.ru/matlab/MatLab_RU/matlab/default.asp.htm
- 29. http://www.nsu.ru/matlab/MatLab_RU/simulink/book2/6.asp.htm
- 30. http://www.nsu.ru/matlab/MatLab_RU/simulink/book2/5.asp.htm
- 31. http://www.nsu.ru/matlab/MatLab_RU/simulink/book2/4.asp.htm
- 32. http://www.nsu.ru/matlab/MatLab_RU/simulink/book2/3.asp.htm
- 33. http://www.nsu.ru/matlab/MatLab_RU/simulink/book2/2.asp.htm
- 34. http://www.nsu.ru/matlab/MatLab_RU/simulink/book2/1.asp.htm

ПРИЛОЖЕНИЯ

Советы пользователям Simulink

1. Использование переменных для задания параметров блоков

Параметры блоков **Simulink** можно задавать не только как числовые параметры, но и как имена переменных. Значения переменных необходимо задать перед началом расчета. Такой способ задания параметров удобен, если требуется проводить расчеты для множества вариантов параметров блоков модели. На рис.1.1 показана схема такой модели. Параметры блока “**Передаточная функция**” заданы переменными **a**, **b** и **c**. Окно параметров блока “**Передаточная функция**” показано на рис. 1.2

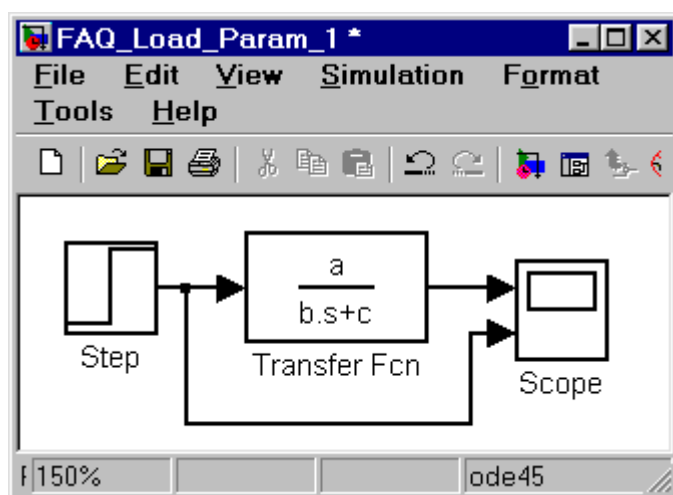


Рис 1.1. Модель, в которой параметры блоков заданы как имена переменных

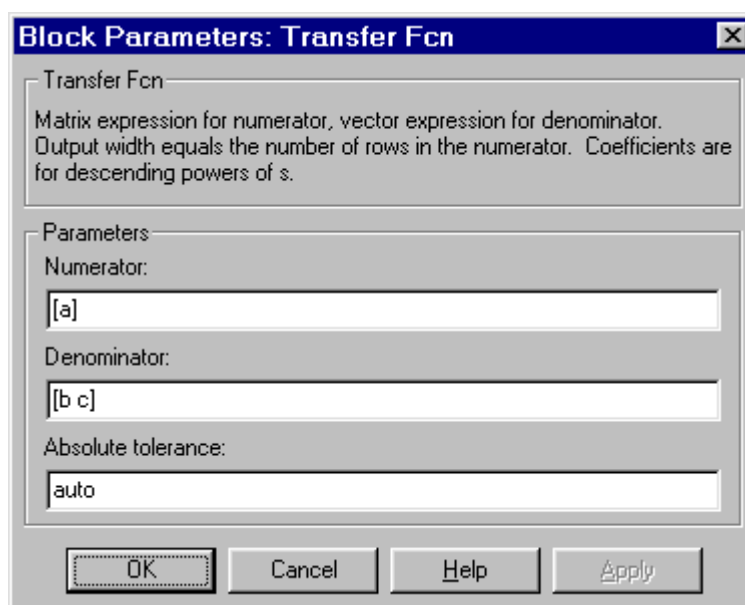


Рис 1.2. Окно параметров блока “Передаточная функция”

Ввод значений переменных можно выполнить из командного окна **MATLAB**. Для этого после приглашения (символ `>>`) необходимо напечатать команды для присвоения переменным значений, например:

```
a = 1  
b = 2  
c = 1
```

После этого можно выполнить расчет.

В том случае, если параметров много, можно создать **m**-файл с командами присвоения переменным значений, открыть его из командного окна **MATLAB** (файл будет открыт в окне отладчика) и запустить его на выполнение командой меню **Debug\Run**.

2. Форматирование графиков в окне блока Scope

Окно блока **Scope** отображающее графики сигналов имеет очень ограниченные возможности по изменению внешнего вида окна и графиков. Фактически можно изменять лишь масштаб отображения графиков, а команды изменения цветов линий и фона окна, толщины линий, размера и типа шрифтов и т.п. недоступны. Чтобы получить доступ к этим возможностям необходимо после проведения расчетов открыть окно блока **Scope** и в основном окне **MATLAB** ввести следующие команды:

```
set(0,'ShowHiddenHandles','On')
```

```
set(gcf,'menubar','figure')
```

После выполнения этих команд в окне блока **Scope** появится меню как это показано на рис. 2.1.

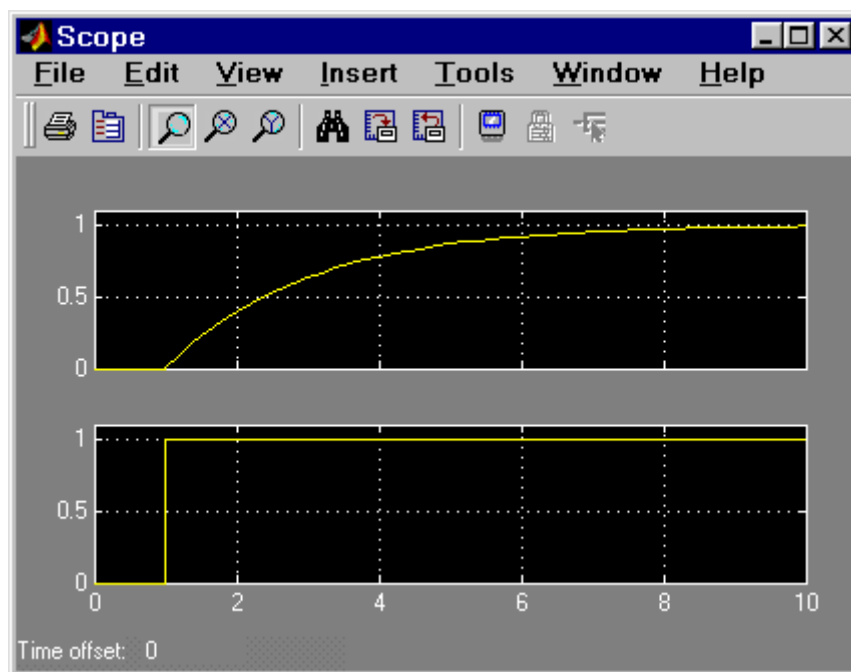


Рис 2.1. Окно блока **Scope**

Используя команды меню **Edit/Figure Properties...**, **Edit/Axes Properties...** и **Edit/Current Object Properties...** можно изменить внешний вид окна, осей и самих графиков в соответствии со своими требованиями. Пример таких изменений показан на рис. 2.2. Изложенное выше относится и к другим блокам, предназначенным для просмотра графиков: **Power Spectral Density**, **Averaging Power Spectral Density**, **Cross Correlator** и т.п.

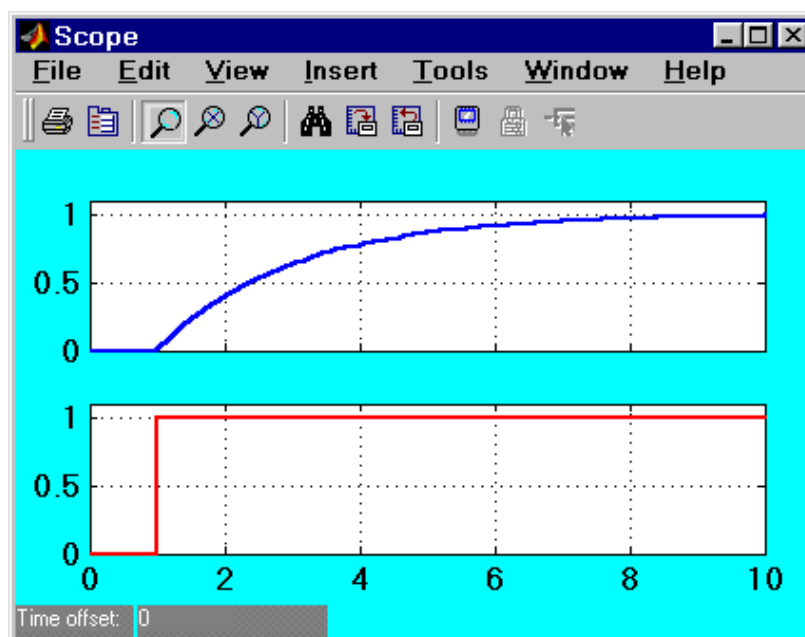


Рис 2.2. Измененное окно блока **Scope**

3. Запись результатов расчета в текстовый файл

В некоторых случаях после проведения расчетов требуется передать результаты в другую программу для их последующей обработки. Такую процедуру удобно осуществить с помощью текстового файла.

Для записи результатов необходимо сформировать вектор необходимых сигналов (включая время), передать этот вектор в рабочее пространство **MATLAB** и, по окончании расчета, записать данные в текстовый файл.

Пример модели реализующей данный принцип показан на рис.3.1. В окне параметров блока **To workspace** необходимо указать имя переменной для вывода в рабочее пространство (в примере это имя - **simout**), а также установить значение параметра **Save format** (формат записи) - **Array** (массив).

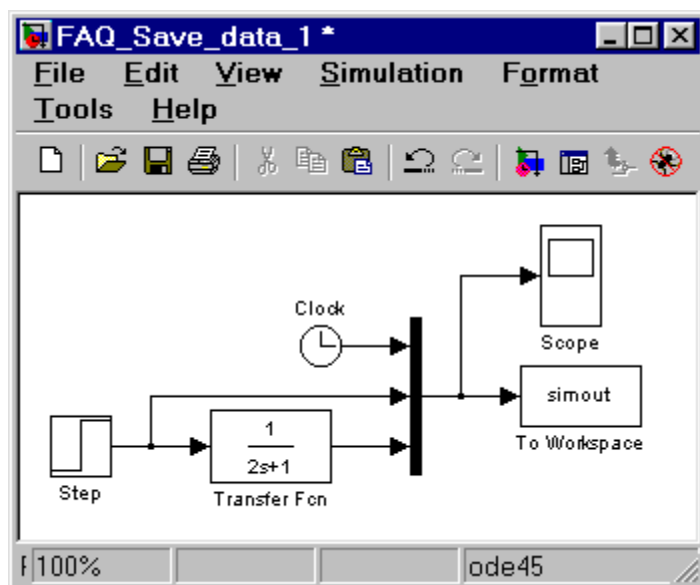


Рис 3.1. Пример модели, использующей вывод результатов расчета в рабочее пространство **MATLAB** для последующей записи в текстовый файл.

Для записи в файл необходимо в командном окне **MATLAB** задать на выполнение команду вида:

```
save('outfile.dat','-ascii','simout','-double')
```

где **outfile.dat** – имя файла для записи,

simout – имя переменной значение которой записывается в текстовый файл

-double – режим записи с удвоенной точностью.

Для реализации процедуры записи переменной из рабочего пространства в файл можно также использовать функцию **dmlwrire**. Однако данная функция записывает результаты с точностью 4 знака после запятой, что может оказаться недостаточно. Целесообразно создать копию этой функции под другим именем (например, **my_dlmwrite**), в которой требуется изменить объявление функции:

```
function my_dlmwrite(filename, m, dlm, r, c, N_digits)
```

и вызов функции **num2str** :

```
str = num2str(m(i,j),N_digits);
```

Параметр **N_digits** будет определять количество значащих цифр после запятой.

Для записи в файл следует в командном окне осуществить вызов функции, например:

```
my_dlmwrite('outfile.dat',simout,' ',0,0,6)
```

В данном примере число значащих цифр задано равным 6.

Вызов новой функции можно осуществлять не только из командного окна **MATLAB**. Можно создать пустую маскированную подсистему и указать для нее функцию записи в файл как функцию открытия (**Edit\Block Properties**, параметр **Open function**).

Сама новая функция должна находиться в текущем рабочем каталоге.

4. Вывод значений вектора в несколько колонок в блоке Display

Блок **Display** обеспечивает вывод значений сигнала в цифровой форме. При этом в окне блока может отображаться как скалярный сигнал (одно значение), так и вектор или матрица. При отображении векторного сигнала большой размерности все значения вектора могут не поместиться на экран. В этом случае можно использовать блоки **Submatrix** из

библиотеки **DSP Blockset** для извлечения из вектора новых векторов с фиксированным числом строк (например, первый блок **Submatrix** извлекает значения с 1 по 20 строку, второй - с 21 по 40 и т.д.) и, затем, подать эти векторы на блок **Matrix Concatenation**, указав горизонтальный метод конкатенации. В результате на **Display** будут выведены значения сигнала в несколько колонок (матрица значений сигнала).

На рис 4.1 показан пример модели реализующий представленную методику. В примере значения вектора из 6 элементов выводится блоком **Display** в две колонки.

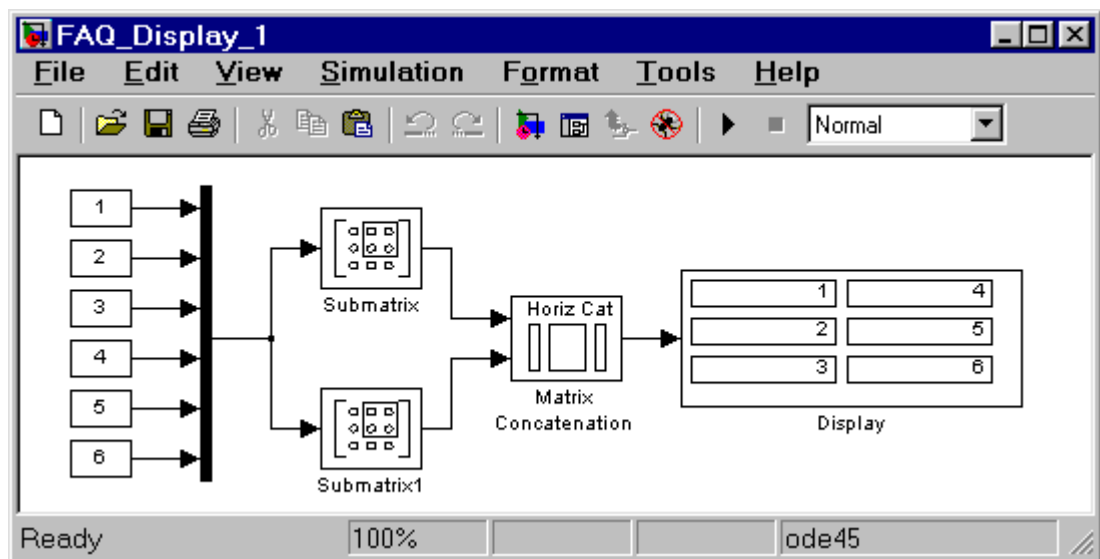


Рис 4.1. Пример модели, использующей вывод значений вектора в несколько колонок

5. Отображение рисунков в окне модели

Для повышения наглядности модели может потребоваться поместить в окне модели рисунок. Стандартные средства для этого отсутствуют.

Для того чтобы обойти это ограничение необходимо:

1. Создать пустую подсистему.
2. Выполнить ее маскирование (**Edit\Mask Subsystem**).
3. Открыть окно редактирования маски(**Edit\Edit Mask...**).
4. На вкладке **Icon** в графе **Drawing Commands** указать команду для считывания из файла и отображения графического образа, например:

```
image(imread('Picture_1.bmp','bmp'))
```

где первый параметр – имя графического файла, а второй – его тип.

На рис. 5.1 показан пример окна модели с созданными таким образом рисунками.

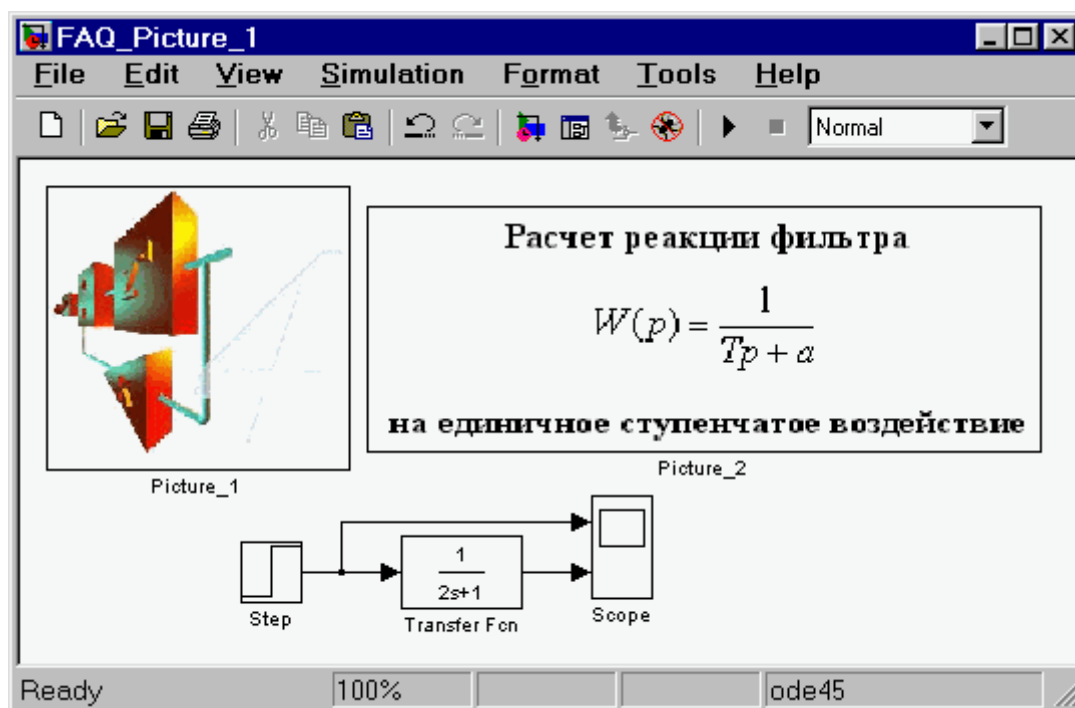


Рис 5.1. Пример модели с рисунками

6. Создание расширенной справки для маскированных подсистем

Редактор маски имеет весьма ограниченные возможности по созданию справочной системы. В окне **Block Help** (справка блока) на вкладке **Documentation** (документация) допускается вводить лишь текстовые сообщения и нет возможности напрямую ввести рисунки, формулы и т.д. К тому же с тексты, написанные на русском языке, могут оказаться не читаемыми. Выход из создавшейся ситуации может быть следующим: создать **HTML**-файл со справочной информацией и в окне **Block Help** записать команду для открытия данного файла справочной системой **MATLAB**. Такая команда может выглядеть следующим образом:

```
web(['file:/// ' which('help_1.htm')]);
```

где **help_1.htm** – имя файла со справочной информацией,

which – команда, возвращающая полный путь указанного в качестве параметр файла.

web – команда, обеспечивающая запуск справочной системы и загрузку в нее нужного файла.

Для корректной работы такой справки необходимо, чтобы файл модели и файл справки находились в одной папке, и данная папка была установлена как рабочая (графа **Current directory** в основном окне **MATLAB**). На рис. 6.1 показана модель с маскированным блоком **Transfer Fcn** и окно справки для этого блока.

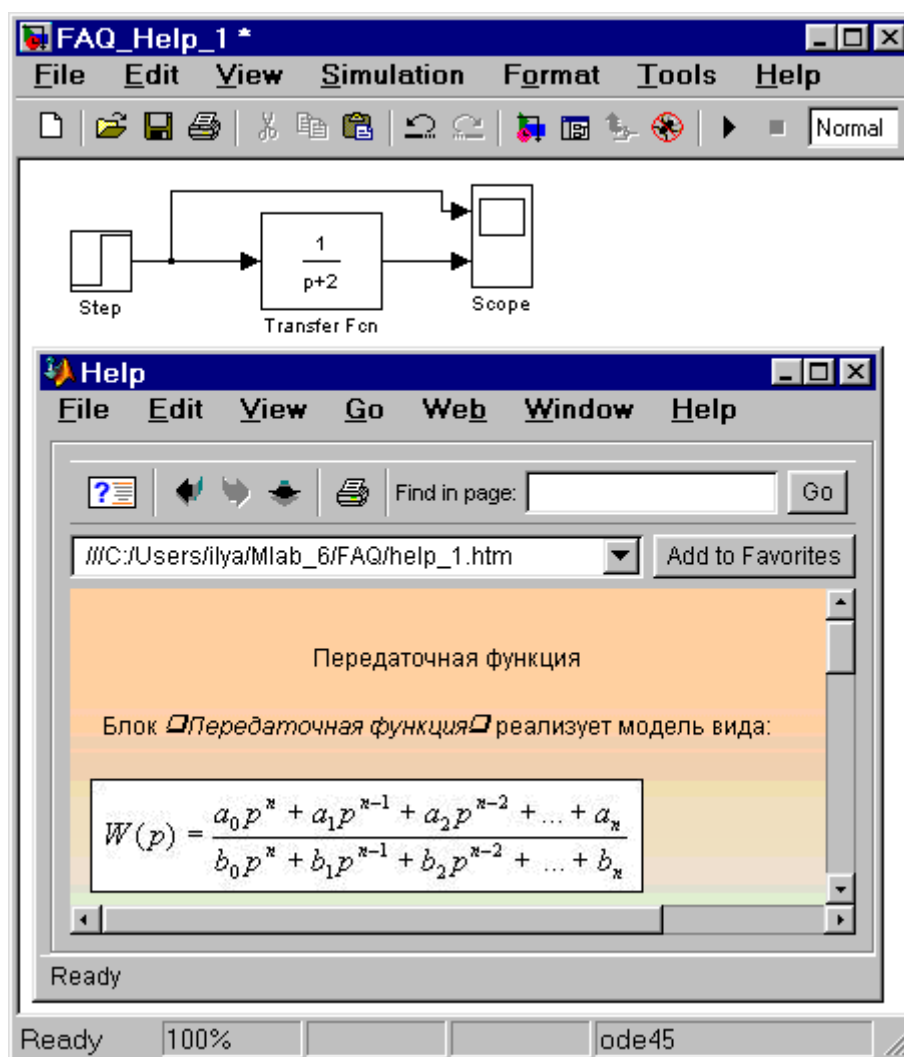


Рис 6.1. Окна модели и справочной системы

7. Нестандартное открытие подсистем

Для повышения удобства работы с моделью команды, которые обычно требуется задавать из основного окна **MATLAB** можно оформить

в виде **m**-файлов (или функций) и запускать на выполнение, используя подсистему. Для реализации такой методики необходимо:

1. Создать пустую подсистему.
2. Оформить нужные команды в виде **m**-файла.
3. В свойствах подсистемы (**Edit/Block Properties...**) указать имя требуемого **m**-файла (графа **Open function**). Окно диалога для установки свойств подсистемы показано на рис. 7.1.

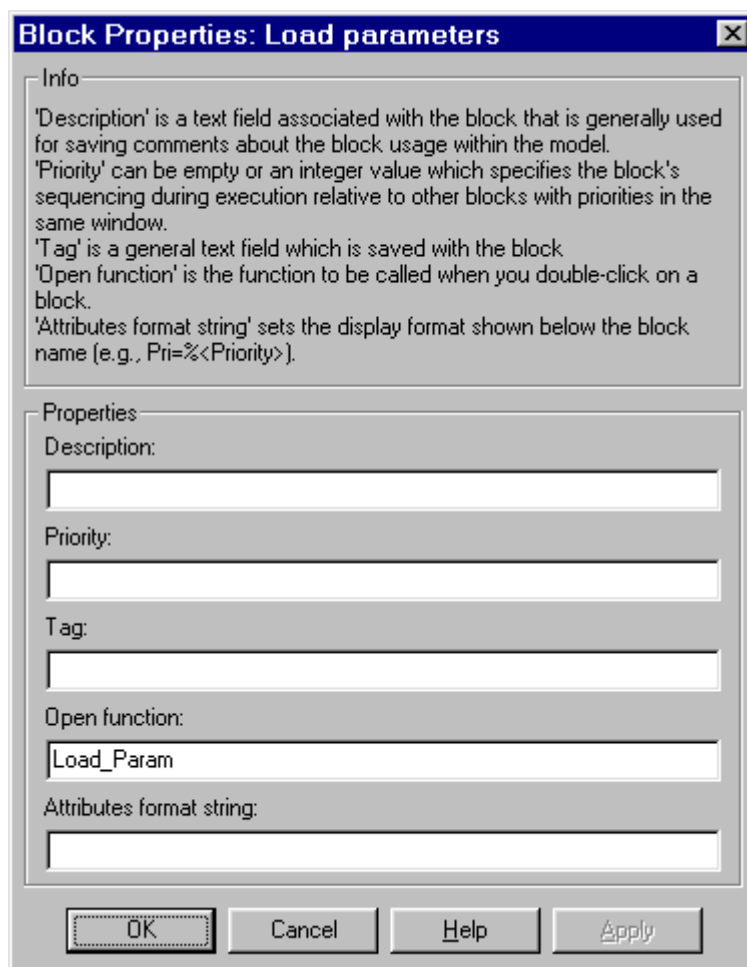


Рис. 7.1 Окно диалога для установки свойств подсистемы.

В приведенном ниже примере (рис. 7.2) подсистема **Load Parameters** использована для загрузки параметров модели. При выполнении двойного щелчка левой клавишей мыши на изображении этой подсистемы происходит выполнение файла **Load_Param.m**:

Phase_A=0;

Phase_B=2*3.1415926/3;

В результате начальным фазам сигналов присваиваются значения из файла.

После проведения расчетов открытие подсистемы **Plot Results** запускает файл **Plot_3D.m**:

```
x=simout(:,1);
```

```
y=simout(:,2);
```

```
z=simout(:,3);
```

```
plot3(x,y,z);
```

С помощью команд, записанных в данном файле, происходит извлечение столбцов из массива **Simout**, передаваемого в рабочее пространство **MATLAB** и построение графика линии в трехмерной системе координат.

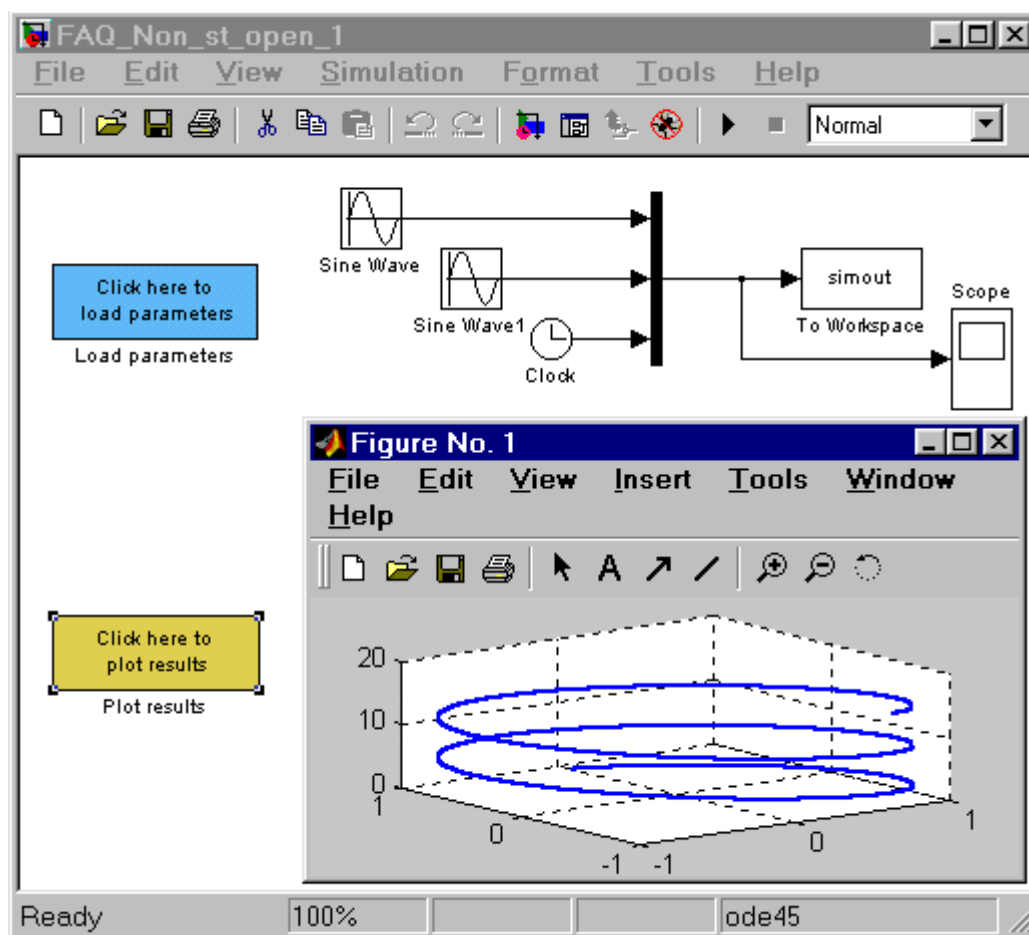


Рис 7.2. Пример модели с подсистемами, использующими нестандартные функции открытия

8. Автоматическое выполнение команд до и после выполнения расчета.

Выполнение **m**-файлов перед началом расчета и после его окончания можно задать с помощью окна установки свойств модели (**File/Model Properties...**). На рис. 8.1 показано данное окно. В графе **Model initialization function** записано имя **m**-файла, выполняющего загрузку параметров модели. В графе **Simulation stop function** указано имя **m**-файла, обеспечивающего построение трехмерного графика после окончания расчета (содержимое файлов, также как и описание модели, дано в предыдущем примере).

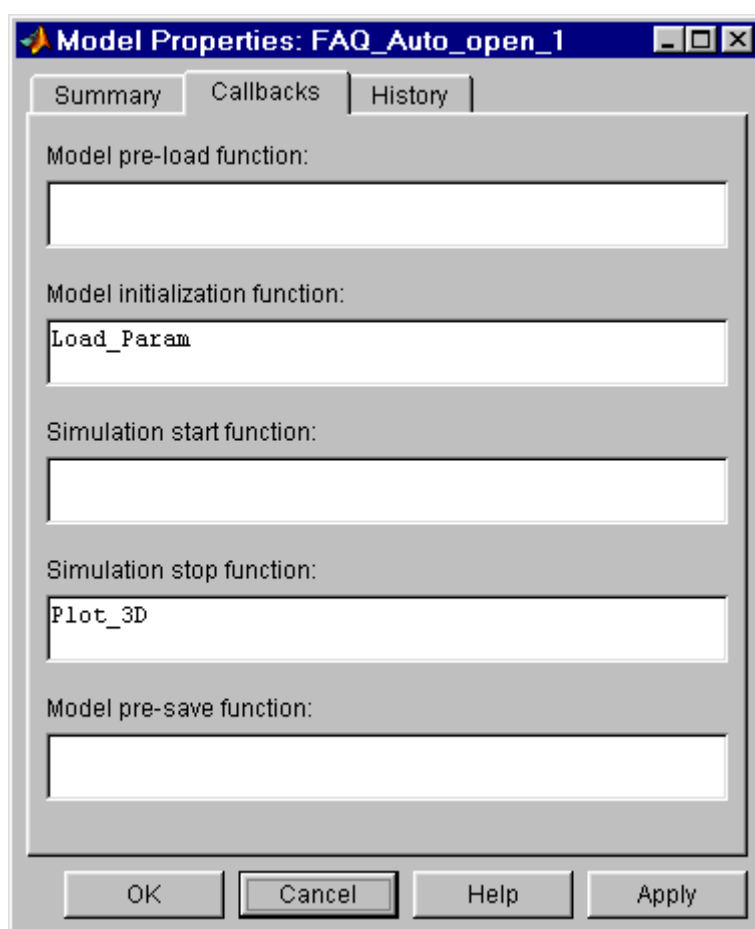


Рис 8.1. Окно свойств модели

9. "Memory allocation error" – Ошибка выделения памяти

При длительных вычислениях с малым шагом расчет может прекращаться с сообщением **"Memory allocation error"**. В том случае, если шаг расчета из-за потери точности увеличить нельзя, можно воспользоваться методикой проведения расчета в несколько сеансов. Для этого необходимо:

1. Открыть окно установки параметров (**Simulation parameters**). На вкладке **Workspace I/O** (параметры взаимодействия с рабочей областью) отметить галочкой параметр **Final State** (состояние системы при окончании расчета), а также задать имя переменной для записи (например, **xFinal**). Окно параметров расчета для этого случая показано на рис. 9.1

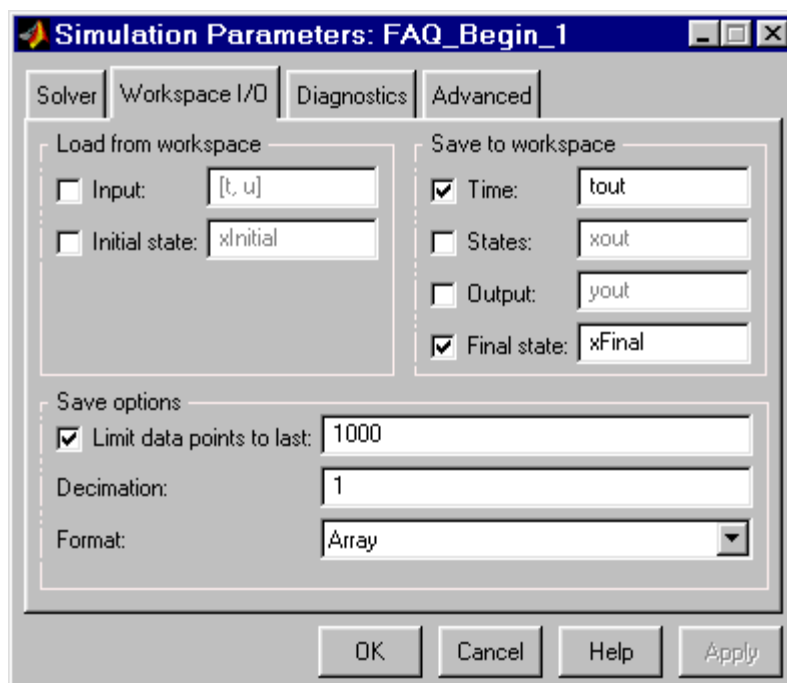


Рис 9.1. Окно параметров расчета на первой стадии

2. Выполнить расчет.

3. Открыть окно установки параметров (**Simulation parameters**). На вкладке **Workspace I/O** отметить галочкой параметр **Initial State** (состояние системы в начале расчета) и указать имя переменной сохранившей результаты первой части расчета (**xFinal**). Окно параметров расчета для этого случая показано на рис. 9.2

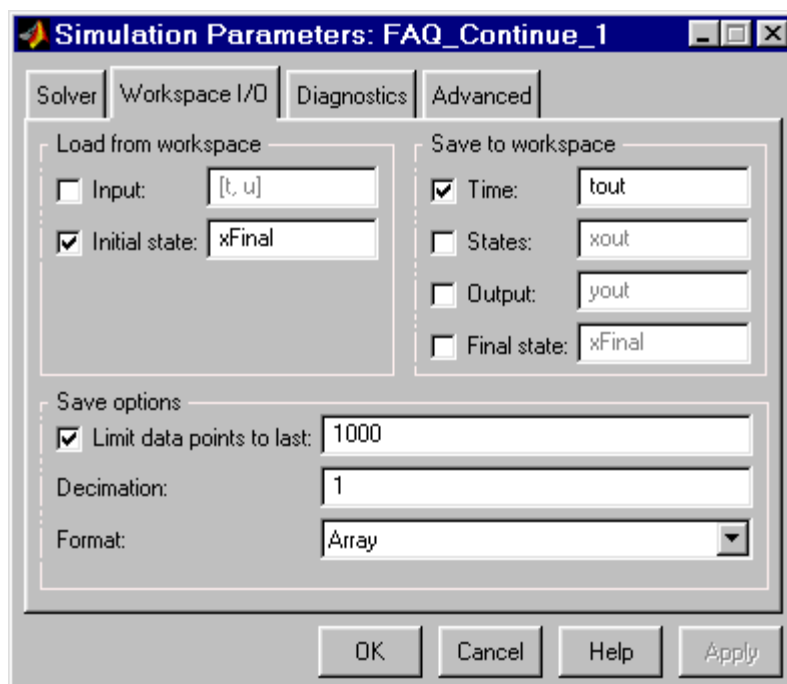


Рис 9.2. Окно параметров расчета на второй стадии

4. Выполнить вторую часть расчета.

На рис. 9.3. показана модель и результаты расчета выполненные в два этапа.

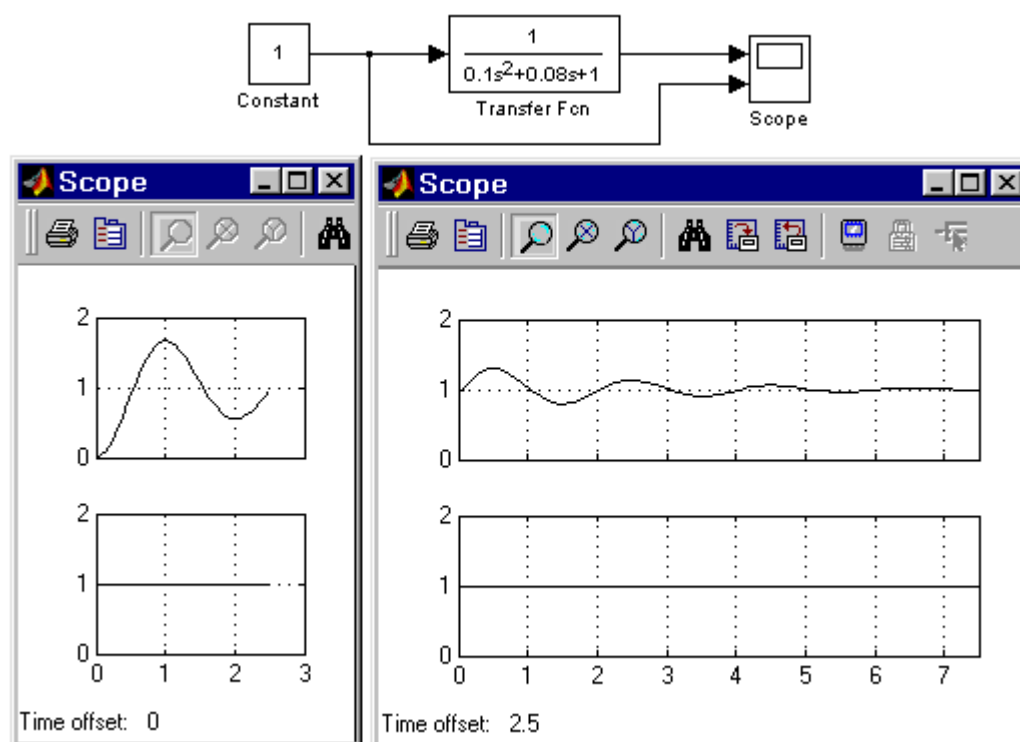


Рис 9.3. Модель и результаты расчета

10. Изменение параметров блока в процессе расчета

В некоторых случаях требуется, чтобы параметр блока задаваемый в окне его свойств, изменялся в процессе расчета (в том числе и по сигналу вырабатываемому другим блоком данной модели). Например, необходимо, чтобы частота источника синусоидального сигнала изменялась в соответствии с выходным сигналом какого-либо блока. Пример такой модели показан на рис. 10.1.

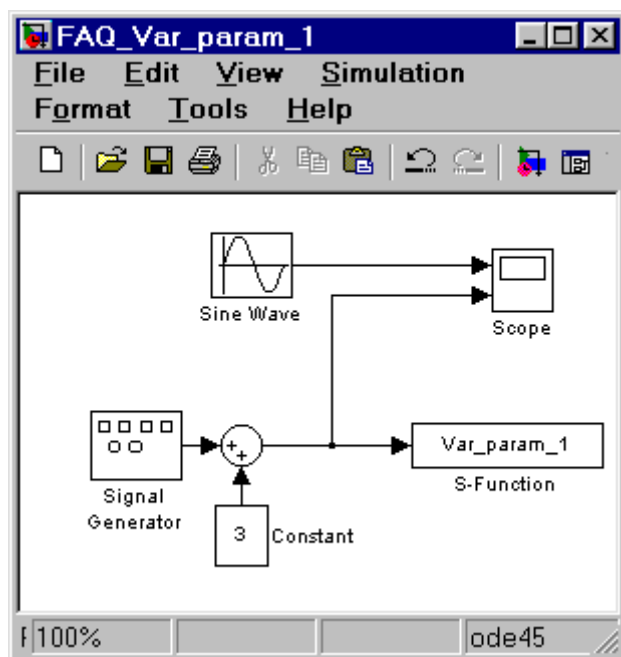


Рис 10.1. Расчетная модель

В рассматриваемой модели блоки **Signal Generator**, **Constant** и **Sum** формируют сигнал задания на частоту блока **Sine Wave**. Данный сигнал на каждом шаге расчета с помощью **S-функции** преобразуется в значение параметра **Frequency** блока **Sine Wave**. Результаты моделирования представлены на рис. 10.2. На верхнем графике показан выходной сигнал синусоидального источника, на нижнем – задание на частоту источника.

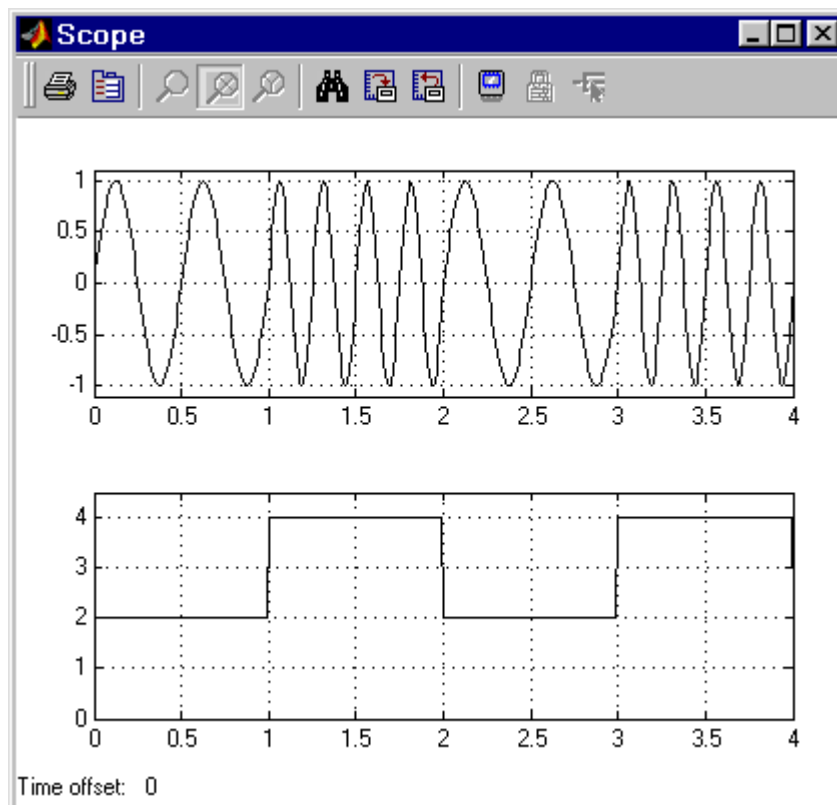


Рис 10.2. Результаты расчета

Текст **S-функции** приведен ниже:

```
function [sys,x0,str,ts] = Var_param_1(t,x,u,flag)
%
switch flag,
case 0,
[sys,x0,str,ts]=mdlInitializeSizes;
case 3,
sys=mdlOutputs(t,x,u);
case {1,2,4,9}
sys=[]; % do nothing
otherwise
error(['Unhandled flag = ',num2str(flag)]);
end
%
function [sys,x0,str,ts]=mdlInitializeSizes
%
sizes = simsizes;
sizes.NumContStates = 0;
```

```

sizes.NumDiscStates = 0;
sizes.NumOutputs = 0;
sizes.NumInputs = 1;
sizes.DirFeedthrough = 1;
sizes.NumSampleTimes = 1;
sys = simsizes(sizes);
x0 = [];
str = [];
ts = [0 0];
%
function sys=mdlOutputs(t,x,u)
set_param('FAQ_Var_param_1/Sine
Wave','Frequency',num2str(u*2*3.1415926))
%
sys = [];

```

Задачу установки нового значения частоты на каждом шаге расчета решает функция **set_param**, в параметрах которой указаны имя файла модели и имя блока, а также название изменяемого параметра блока. Функция **num2str** преобразует числовое значение параметра в символьную строку. Переменной **u** обозначен входной сигнал **S-функции** (значение частоты в Гц с помощью множителя **2*3.1415926** переводится в рад/с).

11. Создание буфера хранящего несколько последних расчетных значений

Для расчета интегральных показателей исследуемого сигнала необходимо иметь несколько значений этого сигнала. Решить поставленную задачу можно с помощью **S-функции**, в которой создается массив для хранения нескольких последних значений. Ниже приводится текст такой выполняющей вычисление суммы последних четырех значений сигнала:

```

function [sys,x0,str,ts] = buffer_1(t,x,u,flag)
switch flag,
case 0, % 0 ==> return sizes
[sys,x0,str,ts]=mdlInitializeSizes;
case 2, % 2 ==> return next discrete state

```

```

sys=mdlUpdate(t,x,u);
case 3, % 3 ==> return outputs
sys=mdlOutputs(t,x,u);
case 9, % 9 ==> Termination
sys=mdlTerminate(t,x,u);
otherwise % all other flags
error(['Unhandled flag = ',num2str(flag)]);
end
function [sys,x0,str,ts]=mdlInitializeSizes
sizes = simsizes;
sizes.NumContStates = 0;
sizes.NumDiscStates = 4; % 4 discrete states to hold the "old" values
sizes.NumOutputs = 1;
sizes.NumInputs = 1;
sizes.DirFeedthrough = 0;
sizes.NumSampleTimes = 0; % at least one sample time is needed
sys = simsizes(sizes);
x0 = zeros(1,4);
str = [];
ts = [];
function sys=mdlUpdate(t,x,u)
sys = zeros(4,1); % initialize sys
sys(4) = x(3); % here we are setting the values for
sys(3) = x(2); % x(4), x(3), x(2), and x(1) respectively, for
sys(2) = x(1); % the NEXT integration step
sys(1) = u;
function sys=mdlOutputs(t,x,u)
sys = sum(x); % output the sum of the past four values
function sys=mdlTerminate(t,x,u)
sys = []; % set []

```

В приведенном тексте функция **mdlUpdate** обеспечивает запись и хранение последних четырех значений сигнала, а функция **mdlOutputs** выполняет вычисление их суммы.