

Джендубаев А.-З. Р.
Алиев И. И.

MATLAB, Simulink и SimPowerSystems в электроэнергетике

Учебное пособие

для студентов, обучающихся по направлению подготовки
140400.62 "Электроэнергетика и электротехника",
профиль "Электроснабжение"

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

**СЕВЕРО-КАВКАЗСКАЯ ГОСУДАРСТВЕННАЯ
ГУМАНИТАРНО-ТЕХНОЛОГИЧЕСКАЯ АКАДЕМИЯ**

Джендубаев А.-З. Р.
Алиев И. И.

MATLAB, Simulink и SimPowerSystems в электроэнергетике

Учебное пособие

для студентов, обучающихся по направлению подготовки
140400.62 "Электроэнергетика и электротехника",
профиль "Электроснабжение"

Черкесск
2014

УДК 000000

ББК 00000
И00

Рассмотрено на заседании кафедры "Электроснабжение".

Протокол № ____ от «__» _____ 2014 г.

Рекомендовано к изданию редакционно-издательским советом СевКавГГТА.

Протокол № ____ от «__» _____ 2014 г.

Рецензенты: Эркенов Н. Х. – к. т. н., доцент кафедры "Электроснабжение"

Джендубаев А.-З. Р.

И00 MATLAB, Simulink и SimPowerSystems в электроэнергетике: учебное пособие для студентов, обучающихся по направлению подготовки 140400.62 "Электроэнергетика и электротехника", профиль "Электроснабжение" / Джендубаев А.-З. Р., Алиев И. И. – Черкесск: БИЦ СевКавГГТА, 2014. – 136 с.

В учебном пособии дано краткое описание системы MATLAB R2014b (академическая версия), системы Simulink – главного пакета расширения MATLAB, реализующего визуально блочное имитационное моделирование, как самого общего, так и конкретного назначения, системы SimPowerSystems Specialized Technology – расширения Simulink, предназначенной для моделирования электроэнергетических и электротехнических устройств. Рассмотрена методика создания программ MATLAB, моделей Simulink и SimPowerSystems. Представлены примеры расчета с использованием этих программных продуктов. Дано краткое описание соответствующих библиотек.

Учебное пособие предназначено для студентов, обучающихся по направлению подготовки 140400.62 "Электроэнергетика и электротехника", профиль "Электроснабжение".

УДК
000000
ББК 00000

© Иванов А. В., 2014
© ФГБОУ ВПО СевКавГГТА,
2014

СОДЕРЖАНИЕ

Введение	7
1. ОСНОВЫ РАБОТЫ С MATLAB, Simulink и SimPowerSystems	7
1.1. MATLAB	7
1.1.1. Запуск MATLAB и работа в режиме диалога.....	7
1.1.2. Простые вычисления в MATLAB.....	7
1.1.3. Работа с комплексными числами.....	9
1.1.4. Формирование векторов и матриц.....	11
1.1.5. Двумерная и трехмерная графика MATLAB.....	15
1.1.6. Работа с редактором m-файлов.....	16
1.1.7. Уловный оператор <i>if...else...else...end</i> . Циклы <i>while...end</i> и <i>while...end</i> . Операторы <i>break</i> , <i>continue</i> , <i>return</i> , <i>pause</i>	19
1.1.8. Решение системы дифференциальных уравнений.....	24
1.1.8.1. Трансформатор.....	24
1.1.8.2. Двигатель постоянного тока.....	26
1.1.8.3. Асинхронный двигатель с короткозамкнутым ротором.....	29
1.2. Simulink	35
1.2.1. Система моделирования Simulink.....	35
1.2.2. Состав библиотеки Simulink.....	35
1.2.3. Создание, редактирование и запуск модели Simulink.....	44
1.3. SimPowerSystems	56
1.3.1. Система моделирования SimPowerSystems.....	56
1.3.2. Создание, редактирование и запуск модели SimPowerSystems.....	57
1.3.3. Состав библиотеки SimPowerSystems.....	63
2. МОДЕЛИРОВАНИЕ ЭЛЕМЕНТОВ, УСТРОЙСТВ И СИСТЕМ	

ЭЛЕКТРОЭНЕРГЕТИКИ.....	78
2.1. Моделирование переходных процессов в индуктивности, конденсаторе и выпрямителе.....	78
2.2. Моделирование трансформаторов.....	95
2.2.1. <i>Однофазный трансформатор.....</i>	<i>95</i>
2.2.1. <i>Трехфазный трансформатор.....</i>	<i>103</i>
2.3. Моделирование асинхронного двигателя с короткозамкнутого ротором	115
2.4. Моделирование трансформаторных подстанций 10/0,4 кВ.....	120
2.5.1. <i>Однотрансформаторная подстанция</i>	<i>120</i>
2.5.2. <i>Двухтрансформаторная подстанция</i>	<i>126</i>
Список литературы	135

Введение

Цель этого учебного пособия – научить студентов использовать современный программный продукт **MATLAB** и её расширения **Simulink** и **SimPowerSystems** при осуществлении расчетов, связанных с лабораторными работами, курсовым и дипломным проектированием, а также научными исследованиями.

1. ОСНОВЫ РАБОТЫ С MATLAB, Simulink и SimPowerSystems

1.1. MATLAB

1.1.1. Запуск MATLAB и работа в режиме диалога

Для запуска программы необходимо на рабочем столе найти ярлык



MATLAB и щелкнуть его два раза. На экране появится основное окно рабочей среды MATLAB (рис. 1.1), которое содержит: строку заголовка (MATLAB R2014b); **Toolstrip** (Панель инструментов); **Command Window** (Окно команд); **Workspace** (Рабочая область); **Current Folder** (Текущая папка); **Command History** (История команд); **Details** (Подробности о файле, выделенном в окне **Current Folder**).

На панели инструментов (**Toolstrip**) располагаются шесть вкладок – **HOME** (Главная), **PLOTS** (Графики), **EDITOR** (Редактор), **APPS** (Приложения), **PUBLISH** (Опубликовать) и **VIEW** (Вид).

1.1.2. Простые вычисления в MATLAB

Более детальное изучение многочисленных инструментов MATLAB будем осуществлять по мере необходимости, а пока приступим к непосредственному её использованию. Предположим, что у нас есть переменные $a = 10$ и $b = 20$, которые необходимо сложить. Для этого в **Command Window** (Окне команд) после символа `>>` вводим каждую переменную на новой строке (в конце первых двух строк ставим знак `;`), т.е. запрет вывода в окно команд) и получаем ожидаемый результат:

```
>> a=10;  
>> b=20;  
>> c=a+b  
c =  
    30.
```

Переменные, их значения и сумму можно набрать в одной строке:

```
>> a=10; b=20; c=a+b  
c =  
    30.
```

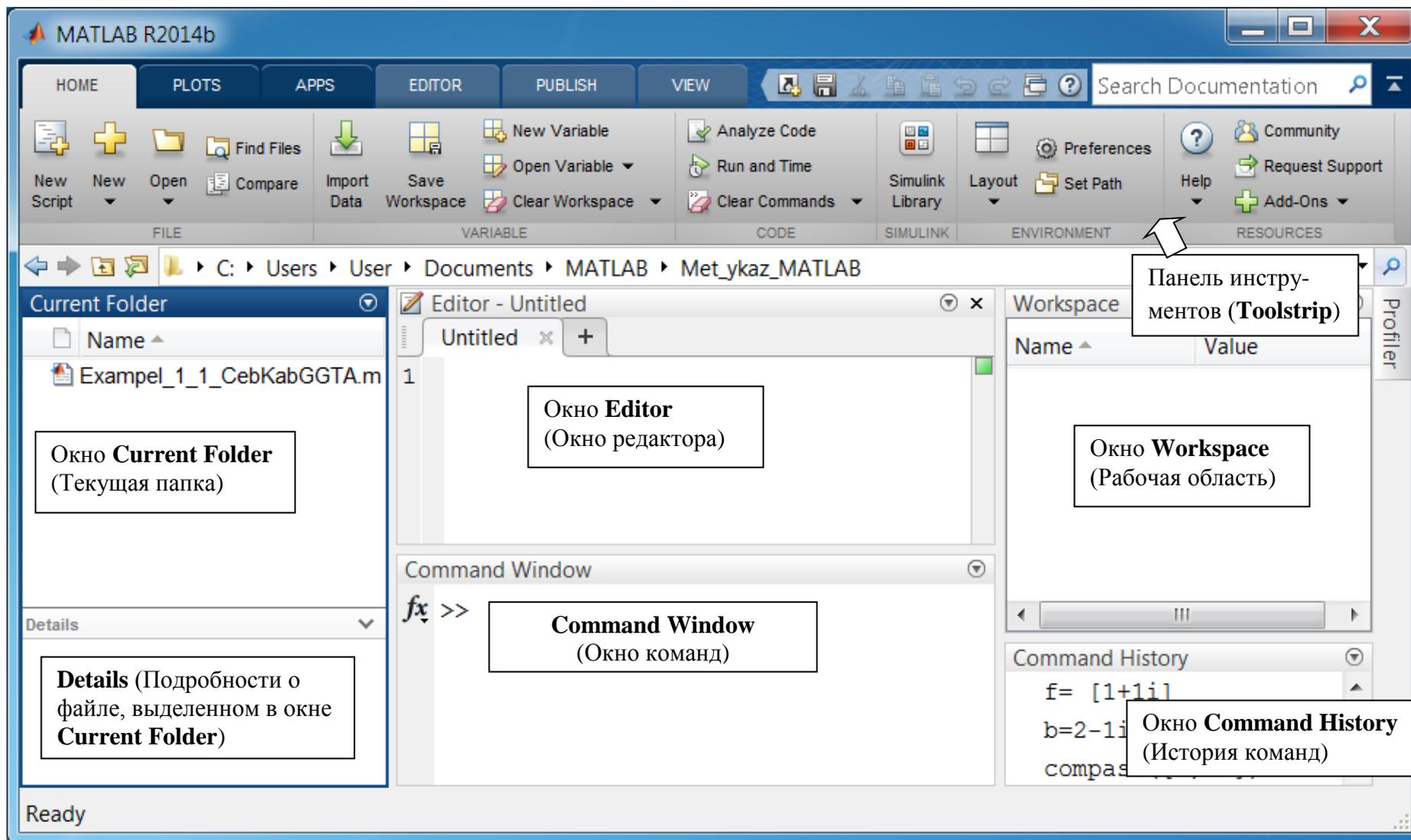


Рис. 1.1

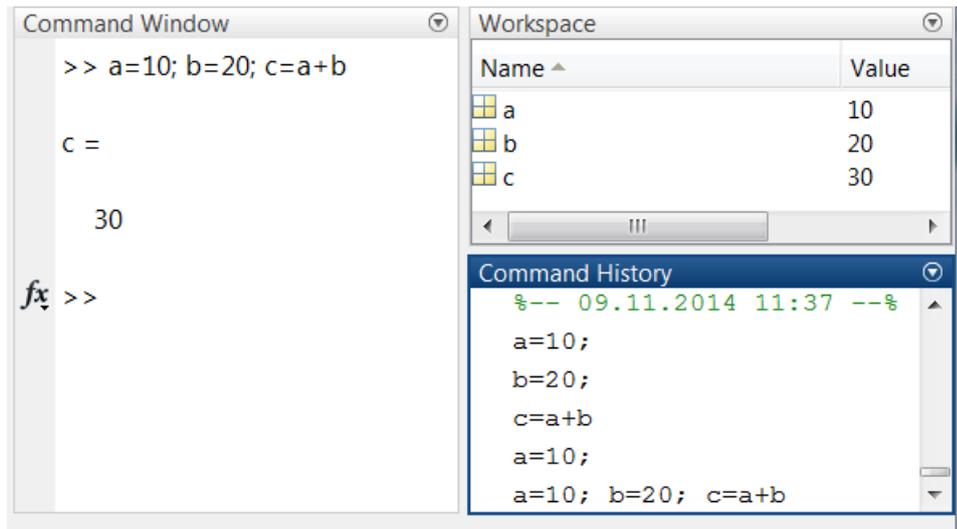
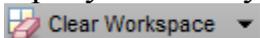
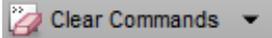


Рис. 1.2

Следует отметить, что информация о переменных автоматически появилась в окне **Workspace** (Рабочая область), а о действиях – в окне **Command History** (История команд) (рис. 1.2). При необходимости в окне **Workspace** переменную можно удалить, изменить её значение (два раза нажать на переменную), можно сохранить (нажать правую кнопку мыши и выбрать **Save**) или очистить окно, нажав на  (Вкладка **HOME**, область инструментов **VARIABLE**).

Используя данные в окне **Command History**, можно вывести в окно **Command Window** любую команду. Для этого достаточно выделить необходимую команду в окне **Command History** и щелкнуть два раза. Аналогичные результаты можно получить, если, находясь в окне **Command Window**, использовать стрелки \uparrow и \downarrow клавиатуры.

Для очистки окна **Command Window** необходимо нажать  (Вкладка **HOME**, область инструментов **CODE**).

1.1.3. Работа с комплексными числами

В **MATLAB** имеется возможность вести расчеты с комплексными числами, которые очень часто используются в электротехнике и электроэнергетике. Для обозначения мнимой единицы используются два зарезервированных имени – **i** и **j**. Более приемлемым вариантом является **i**, т.к. он используется **MATLAB** при выводе результатов [1 , 2 , 3 , 4]:

```

>> a=1+2i; b=3+4i; c=a+b
c =
  4.0000 + 6.0000i

```

Пример 1.1

Определить установившийся ток короткого замыкания однофазного трансформатора (комплексное и абсолютное значения), построить векторную диаграмму и определить фазовый сдвиг между током и напряжением. Напряжение сети: 220 В. Вектор напряжения направлен по мнимой оси. Параметры короткого замыкания трансформатора: $R_k = 2 \text{ Ом}$; $X_k = 3 \text{ Ом}$.

Решение. Первая часть задачи решается достаточно просто (Рис. 1.3):

```
>> I_kz=(0+220i)/(2+3i)
I_kz =
 50.7692 +33.8462i
```

Вторая требует вызова соответствующей функции. Для этого слева от знака >> нажимаем fx . Открывается список категорий, по которым систематизированы функции (Рис. 1.3).

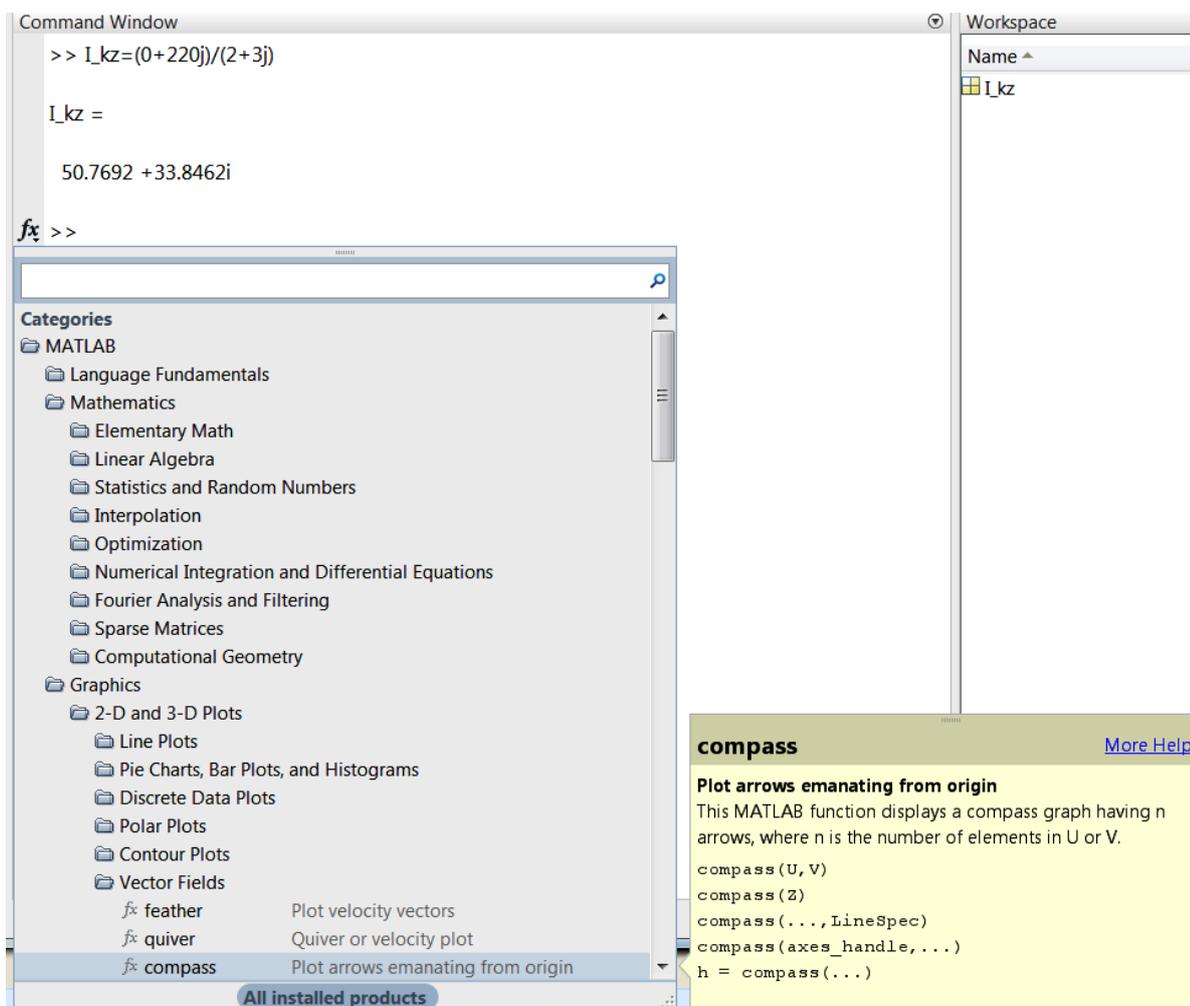


Рис. 1.3

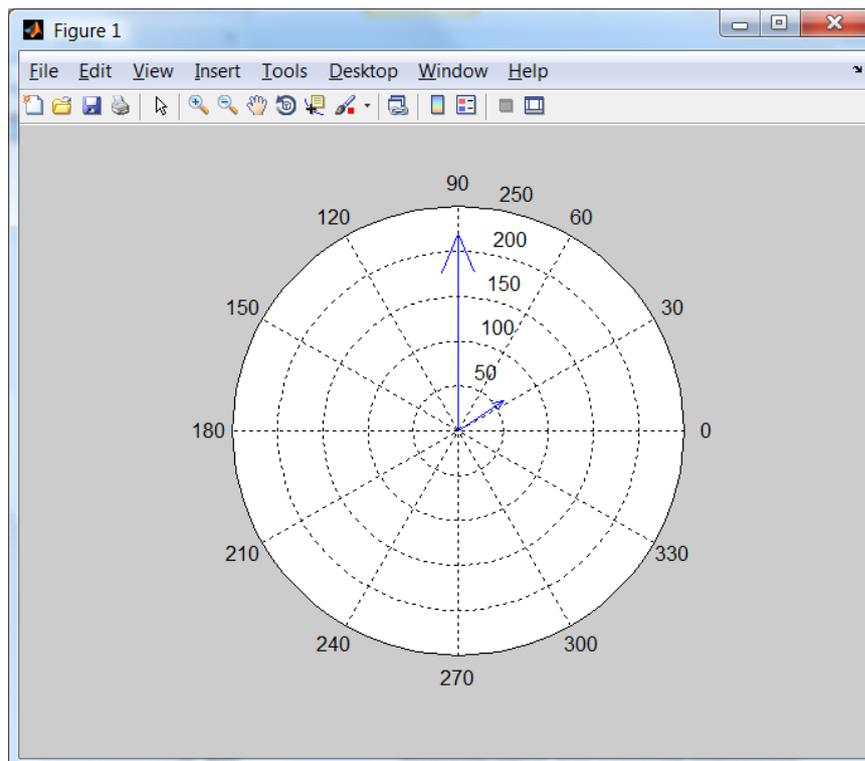


Рис. 1.4

В данном случае воспользуемся функцией **compass**, которая после двух щелчков вставляется в **Command Window**. При необходимости можно посмотреть пример использования этой функции. Для этого переходим на правое окно **compass** и нажимаем **More Help...** Результат построения представлен на рис. 1.4.

Для определения абсолютного значения тока и фазового сдвига между напряжением и током воспользуемся функциями **abs** и **angel** категории **Complex Numbers** (Рис. 1.5).

Следует отметить, что в окне **Command Window** данные можно выделять и копировать (Ctrl+C), однако вставлять можно только после знака **>>**. Окончательный листинг программы *примера 1.1* представлен на рис. 1.6.

1.1.4. Формирование векторов и матриц

Простейший способ задания вектора и матрицы:

```
>> a=[1 2 3 4]
a =
    1    2    3    4
>> b=[1 2 3; 4 5 6; 7 8 9]
b =
    1    2    3
    4    5    6
    7    8    9.
```

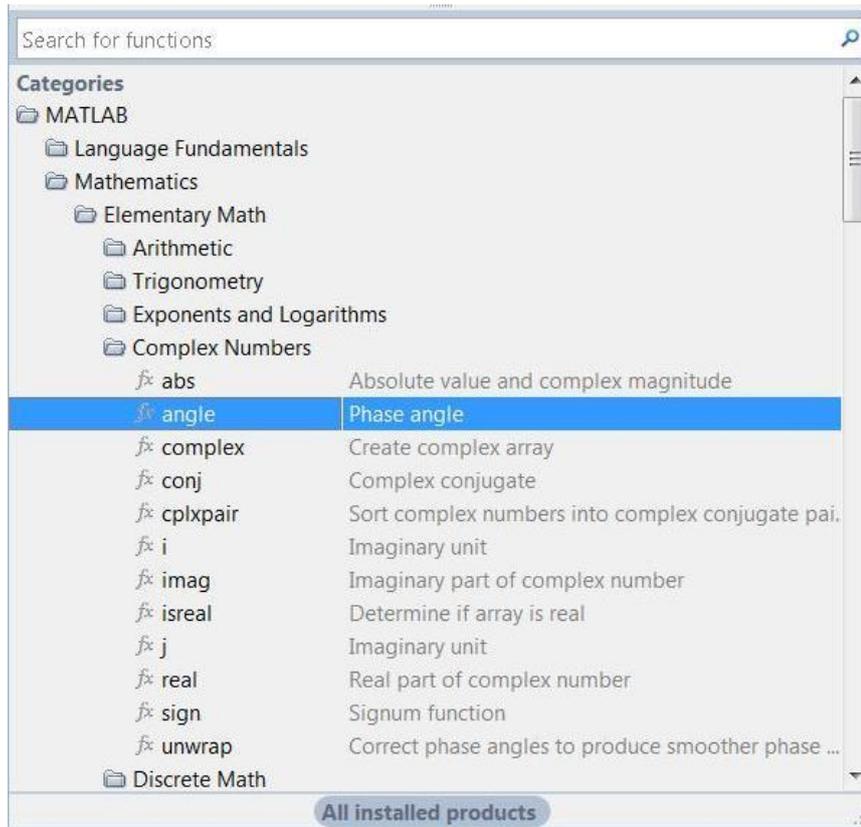


Рис. 1.5

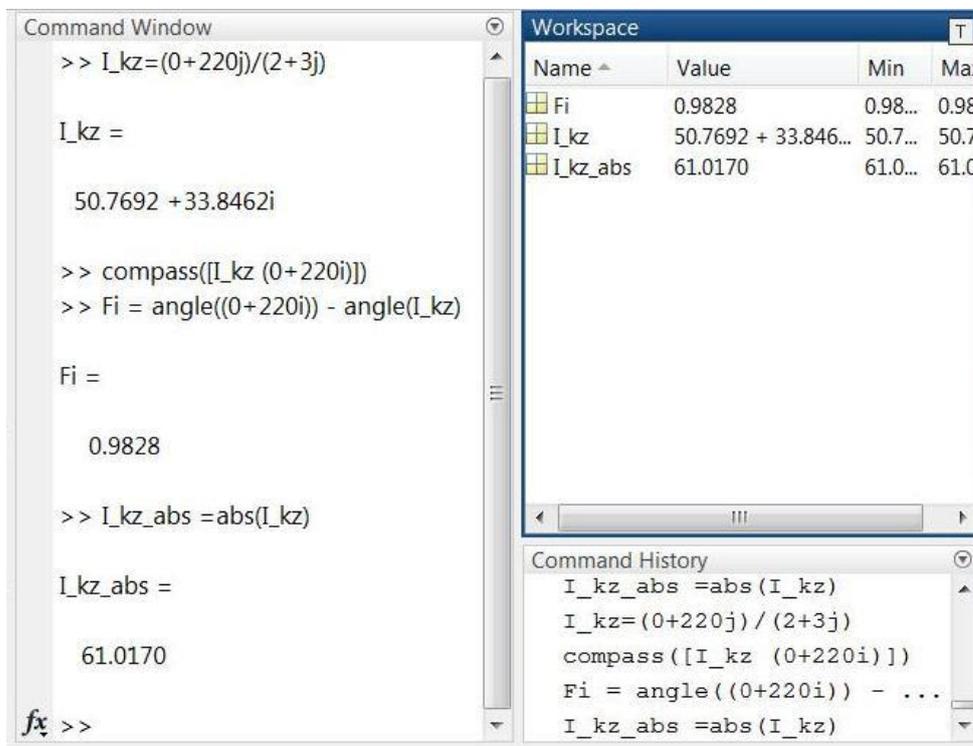


Рис. 1.6

Обращение к элементам матрицы (вектора):

```
>> b(2,3)
ans =
     6
```

Изменение значения элемента:

```
>> b(2,3)=100
b =
     1     2     3
     4     5    100
     7     8     9
```

Элементы вектора и матрицы можно разделять запятой:

```
>> c=[6+5^2, 4-42i, 1+4+3i]
c =
 31.0000 + 0.0000i   4.0000 -42.0000i   5.0000 + 3.0000i.
```

Вектор можно задать следующим способом:

```
>> x=1:2:12
x =
     1     3     5     7     9    11.
```

Здесь шаг равен 2. Если шаг равен 1, то

```
>> x=1:12
x =
     1     2     3     4     5     6     7     8     9    10    11    12.
```

Сумма и разность двух векторов:

```
>> x=1:3, y=4:6, x+y, x-y
x =
     1     2     3
y =
     4     5     6
ans =
     5     7     9
ans =
    -3    -3    -3.
```

Для транспонирования вектора применяется апостроф ('):

```
>> x'
ans =
     1
     2
     3.
```

Скалярное произведение двух векторов (поэлементное умножение):

```
>> x.*y
ans =     4    10    18
```

Поэлементное возведение в степень:

```
>> x.^3
ans =
     1     8    27
```

Поэлементное деление матриц (деление элементов первой матрицы на элементы второй и наоборот):

```
>> x=[1 2 3; 4 5 6]; y=[7 8 9; 10 11 12]; x./y, x.\y
ans =
    0.1429    0.2500    0.3333
    0.4000    0.4545    0.5000
ans =
    7.0000    4.0000    3.0000
    2.5000    2.2000    2.0000.
```

Транспонирование матрицы:

```
>> x'
ans =
     1     4
     2     5
     3     6
```

Обращение матрицы (обратная матрица):

```
>> A=[1 -2 3; 1 1 -8; 2 -6 -3]
A =
     1     -2     3
     1     1    -8
     2     -6    -3
>> I=inv(A)
I =
    1.0408    0.4898   -0.2653
    0.2653    0.1837   -0.2245
    0.1633   -0.0408   -0.0612
```

Решение системы линейных уравнений:

$$x_1 - 2x_2 + 3x_3 = 10$$

$$x_1 + x_2 - 8x_3 = 9$$

$$2x_1 - 6x_2 - 3x_3 = -7$$

```
>> A=[1 -2 3; 1 1 -8; 2 -6 -3]; b=[10;9;-7];
>> x_1=A\b % Первый вариант
>> x_2=inv(A)*b % Второй вариант
x_1 =
    16.6735
     5.8776
     1.6939
```

```
x_2 =  
    16.6735  
     5.8776  
     1.6939
```

1.1.5. Двумерная и трехмерная графика MATLAB

Вывод синусоидальной зависимости с помощью функции MATLAB `plot(x,y)`:

```
>> x=0: pi/100: 4*pi; y=sin(x); plot(x,y).
```

Результат построения представлен на рис. 1.7.

Аналогичный результат можно получить иным способом. Для "чистоты" построения удаляем содержимое окна **Workspace**. С этой целью переходим к этому окну, нажимаем правую кнопку и в выпадающем меню выбираем **Clear Workspace**. После этого повторно создаем вектора:

```
>>x=0: pi/100: 4*pi; y=sin(x).
```

Нажимаем клавишу **Ctrl**, переходим в окно **Workspace**, щелчком выделяем вектора **x** и **y**. Переходим во вкладку **Plots** (Рис. 1.1), нажимаем **Plot** и получаем синусоидальную зависимость, представленную на рис. 1.7.

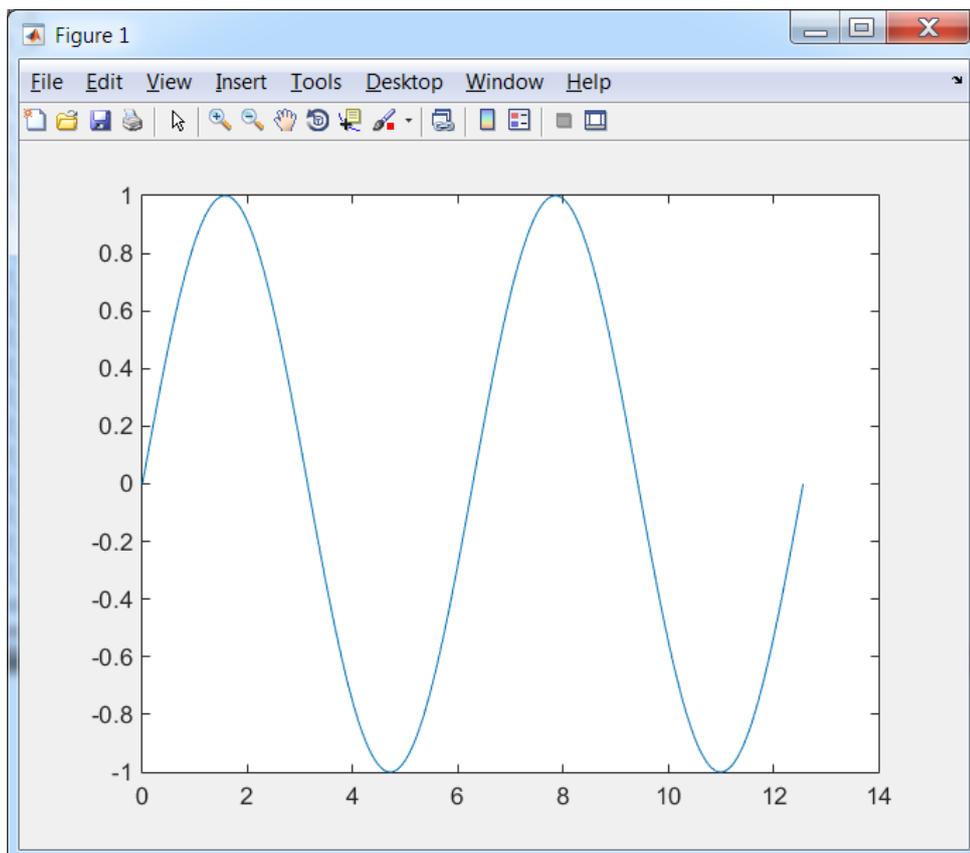


Рис. 1.7.

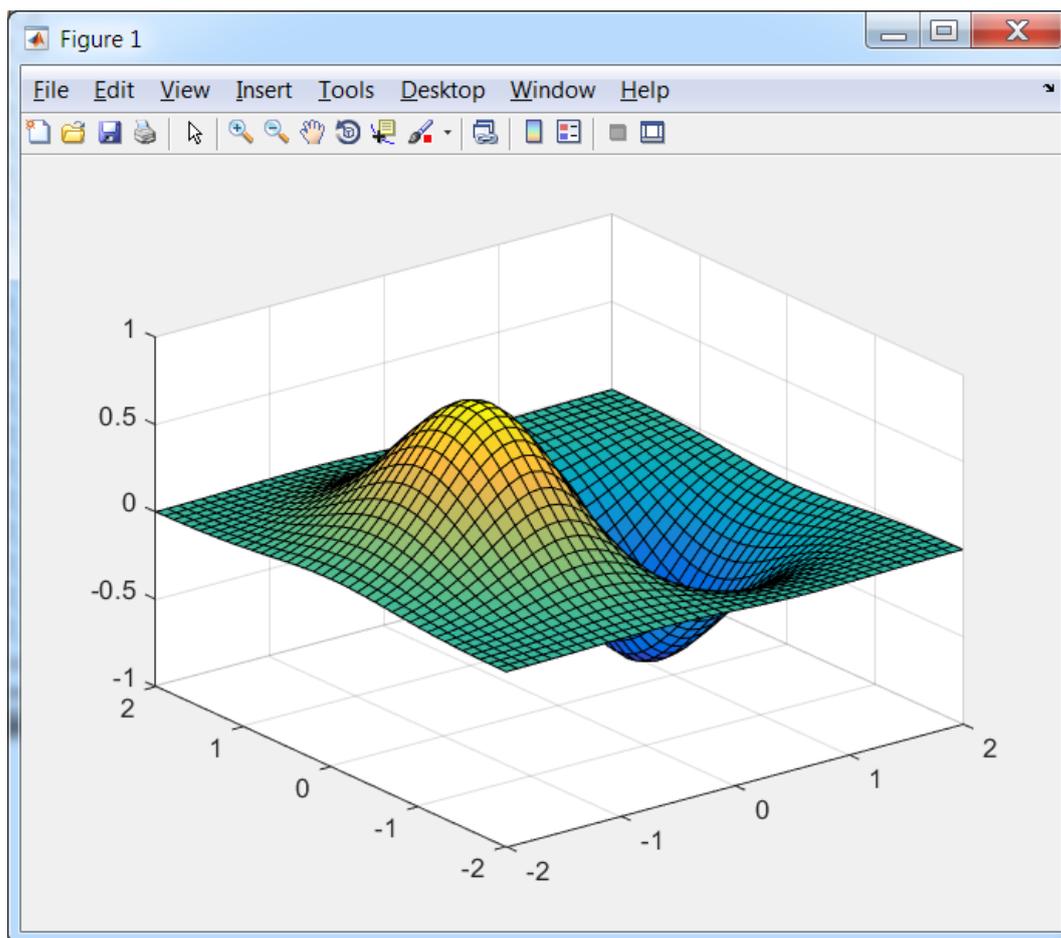


Рис. 1.8

Трехмерный график можно вывести, задав следующие команды:

```
>> x=-2:0.1:2;
>> [X, Y]=meshgrid(x);
>> Z=-2*X.*exp(-X.^2-Y.^2);
>> mesh(X, Y, Z)
>> surf(X, Y, Z) .
```

В результате получим график, представленный на рис. 1.8.

Как и в случае построения двухмерного графика, можно воспользоваться вкладкой **Plots**. Для этого в окне **Workspace** выделим вектора **X**, **Y**, **Z**. После этого активизируется вкладка построения графиков **Plots**, в которой можно выбрать интересующий тип трехмерного графика, в том числе и графика, представленного на рис. 1.8.

1.1.6. Работа с редактором m-файлов

Любую последовательность команд можно оформить в виде m-файлов двух типов: **Script M-Files** (файл-программы) и **Function M-Files** (файл-функции). Для создания файла первого типа необходимо во вкладке **HOME** нажать на **New Script**. При этом открывается окно

редактирования безымянного m-файла (**Untitled**), который необходимо сохранить под другим, удобным для поиска, именем. Если окно редактора **EDITOR** уже открыто, то можно нажать **New** и выбрать **Script** или просто нажать **+**. Имя файла необходимо указывать латинскими буквами без пробелов. Использование кириллицы может привести к возникновению проблем. Нажимаем **Save** → **Save As...** во вкладке **EDITOR** и сохраняем файл, например, под именем *Example_1_1*. В окне введем формулу примера 1.1 и команду построения графика, изображенного на рис. 1.4. Если информация о командах имеется в окне истории, то можно её скопировать и перенести в окно **EDITOR**. После этого нажимаем на стрелку **Run** или на клавишу **F5** и запускаем **Script** (программу) *Example_1_1*. Результаты её работы представлены на рис. 1.9.

Для создания файла второго типа необходимо во вкладке **HOME** нажать на **New** → **Function**. При этом в окне редактора появится текст шаблона функции, который необходимо модифицировать под свои задачи (Рис. 1.10). Предположим, что необходимо создать функцию, которая в качестве входных параметров (**input...args**) получает: напряжение \dot{U} , активное r и реактивное x сопротивление, в качестве выходных (**output...args**) – ток короткого замыкания \dot{I}_{kz} , абсолютное значение тока $I_{kz.abs}$ и угол φ между напряжением и током.

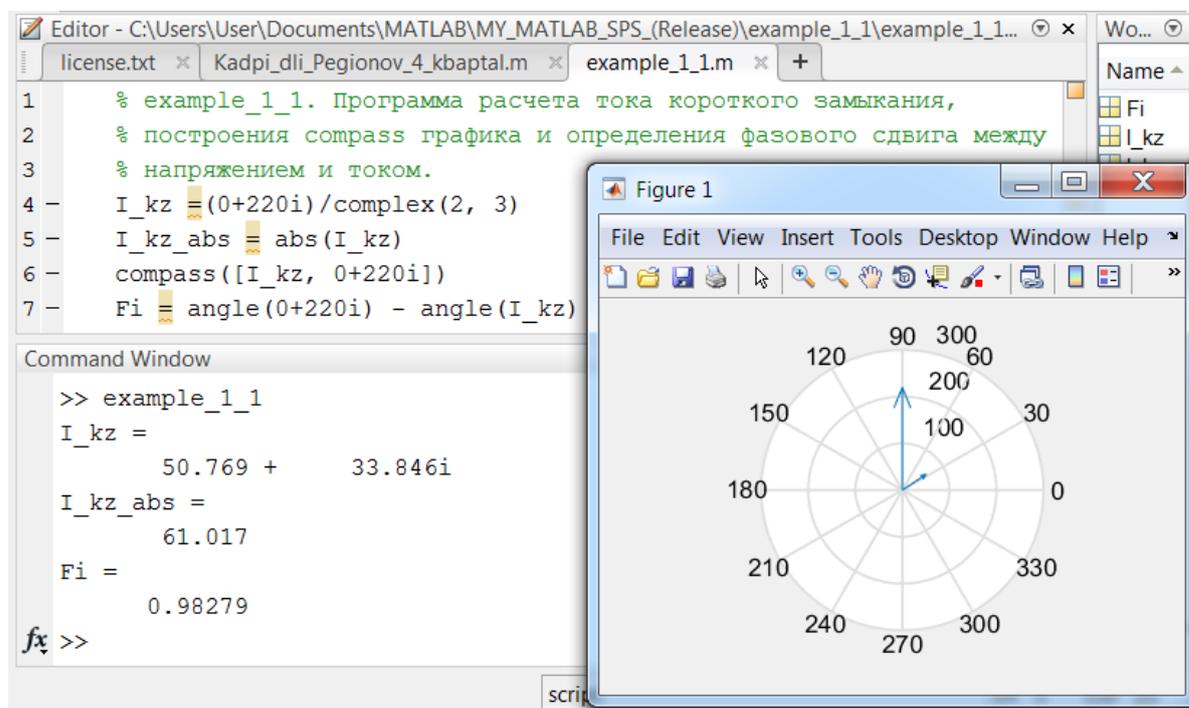
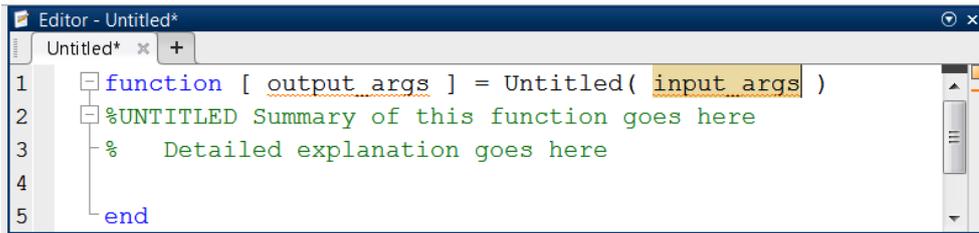


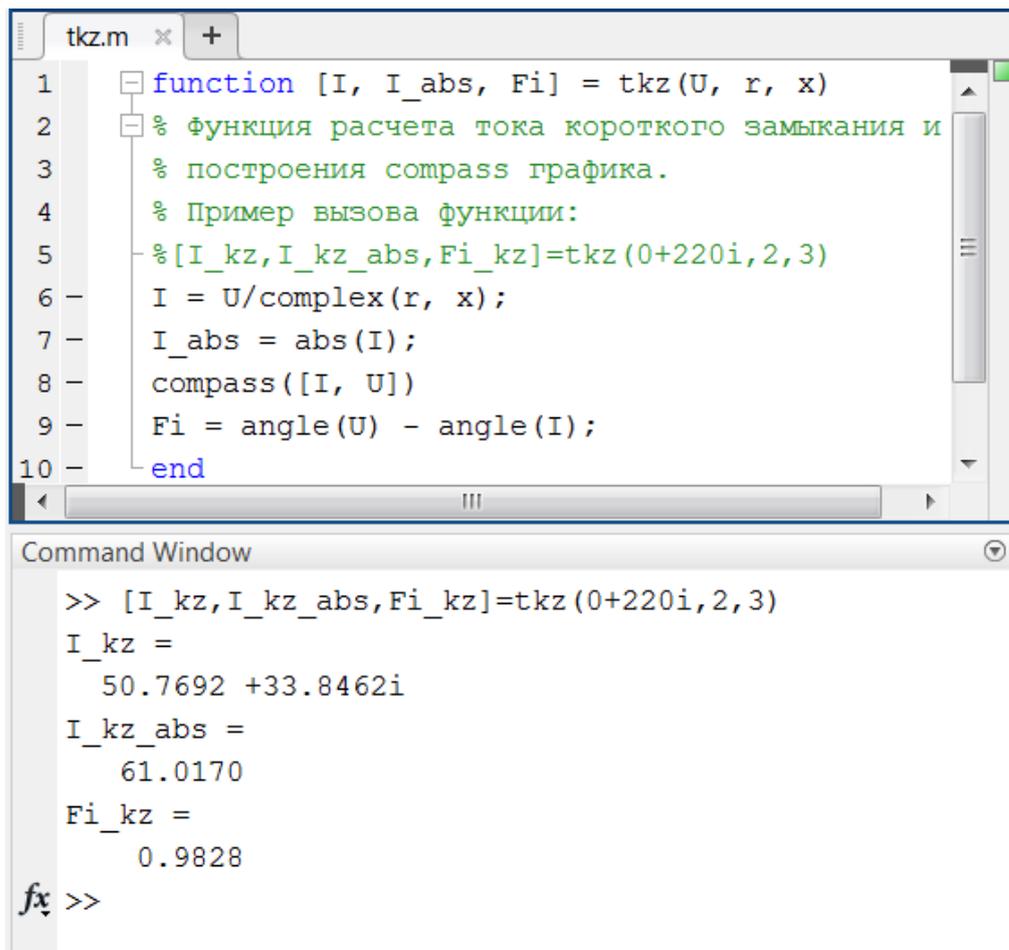
Рис. 1.9



```
Editor - Untitled*
Untitled* x +
1 function [ output_args ] = Untitled( input_args )
2 %UNTITLED Summary of this function goes here
3 % Detailed explanation goes here
4
5 end
```

Рис. 1.10

При этом функция должна построить векторную диаграмму (compass) напряжения и тока. Назовем эту функцию, например, **tkz** и сохраним её в файле с таким же названием, т.е. **tkz**. Совпадение названия функции с названием файла является обязательным условием в **MATLAB**, т.к. при вызове функции из командного окна (**Command Window**) осуществляется вызов файла, в котором дано описание этой функции. Текст функции, её вызов из командного окна представлен на рис. 1.11.



```
tkz.m x +
1 function [I, I_abs, Fi] = tkz(U, r, x)
2 % функция расчета тока короткого замыкания и
3 % построения compass графика.
4 % Пример вызова функции:
5 % [I_kz, I_kz_abs, Fi_kz] = tkz(0+220i, 2, 3)
6 I = U/complex(r, x);
7 I_abs = abs(I);
8 compass([I, U])
9 Fi = angle(U) - angle(I);
10 end

Command Window
>> [I_kz, I_kz_abs, Fi_kz] = tkz(0+220i, 2, 3)
I_kz =
    50.7692 + 33.8462i
I_kz_abs =
    61.0170
Fi_kz =
    0.9828
fx >>
```

Рис. 1.11

Следует отметить, что функцию можно вызвать из *m*-файла, как и любую другую функцию **MATLAB**. Если необходима информация о функции, достаточно в командном окне набрать **help** и название функции, например, **help tkz** или **help sin**.

1.1.7. Условный оператор if...else...else...end. Циклы *mu*na for...end и while...end. Операторы break, continue, return, pause.

Известно, что согласно седьмому изданию ПУЭ при переменном трёхфазном токе: шина *A* имеет желтый цвет, шины *B* и *C* – зелёный и красный соответственно. Нулевой рабочий (нейтральный) проводник *N* обозначается голубым цветом.

Пример 1.2.

Используя условный оператор **if...else...else...end** и оператор цикла **for...end**, создать программу *Example_1_2*, которая по обозначению шины определяет её цвет.

Решение:

```
% Example_1_2. Программа определения цвета шины по её
обозначению.
for m = 1:10 % Организация цикла
    %input - оператор ввода, 's'- для ввода строки символов,
    temp = input(' Введите название шины (A; B; C; N): ', 's');
    if temp == 'A';
        %disp - оператор вывода
        disp('Шина фазы "A" обозначается желтым цветом. ');
    else if temp == 'B';
        disp('Шина фазы "B" обозначается зеленым цветом. ');
    else if temp == 'C';
        disp('Шина фазы "C" обозначается красным
цветом. ') % В m-файле текст между '...' указать одной строкой.
    else if temp == 'N';
        disp('Нулевой рабочий
(нейтральный) проводник "N" обозначается голубым цветом. ')
    else
        disp(' Уточнить раскладку
клавиатуры, регистр или букву. ')
        disp(' Для выхода из цикла
необходимо нажать "Ctrl + C". ')
    end
end
end
end
end
end
```

Для создания цикла можно воспользоваться не только оператором **for...end**, но и оператором **while...end**. Рассмотрим его работу на *Примере 1.3*.

Пример 1.3.

Рассчитать внешнюю характеристику трансформатора ТМ-1000/10 при подключении к нему активно-индуктивной нагрузки с $\cos \varphi_{\text{нагр}} = 0,8$.

Таблица 1.1. Основные технические характеристики трансформатора

Тип, $S_{2H}, \text{кВА}$	$U_{1H},$ кВ	$U_{2H},$ кВ	$P_{xx},$ кВт	$P_{\kappa},$ кВт	$I_{xx},$ $\%$	$u_{\kappa},$ $\%$	Схема, группа соединения обмоток
ТМ 1000	10	0,4	1,9	12,2	1,7	5,5	Y/YH-0

Решение.

На основании данных, представленных в таблице 1.1, определим параметры короткого замыкания трансформатора:

$$I_{2H} = \frac{S_{2H}}{\sqrt{3}U_{2H}} = \frac{1000}{\sqrt{3} \cdot 0,4} = 1443 \text{А} \quad \text{— номинальный фазный ток вторичной}$$

обмотки;

$$z_{\kappa} = \frac{U_{2H} \cdot u_{\kappa} \%}{\sqrt{3}I_{2H} 100\%} = \frac{400 \cdot 5,5\%}{\sqrt{3} \cdot 1443 \cdot 100\%} = 0,0088025 \text{Ом} \quad \text{— полное сопротивление}$$

короткого замыкания;

$$r_{\kappa} = \frac{P_{\kappa}}{3I_{2H}^2} = \frac{12000}{3 \cdot 1443^2} = 0,00192 \text{Ом} \quad \text{— активная составляющая полного}$$

сопротивления короткого замыкания;

$$x_{\kappa} = \sqrt{z_{\kappa}^2 - r_{\kappa}^2} = \sqrt{0,0088025^2 - 0,00192^2} = 0,00859 \text{Ом} \quad \text{— реактивная}$$

составляющая;

$Z_{\kappa} = r_{\kappa} + x_{\kappa}i = 0,00192 + 0,00859i$ — полное сопротивление короткого замыкания в комплексной форме, здесь i — символ для обозначения мнимой части (в **MATLAB** он используется по умолчанию).

Занесем значения линейного напряжения вторичной обмотки в режиме холостого хода, сопротивление короткого замыкания трансформатора и диапазон изменения сопротивления нагрузки. При расчете сопротивление нагрузки с каждой итерацией уменьшается и составляет 80% от предыдущего значения, т.е. $Z_{\text{нагр}}(m) = 0.8 * Z_{\text{нагр}}(m-1)$ (12-я строка m-файла).

```
% example_1_3. Программа расчета внешней
% характеристики трансформатора ТМ-1000/10
clear; U_2_XX = 0+400i; Z_k = 0.00192 + 0.00859i;
Z_nagp_max = 12 + 9i;
Z_nagp(1) = Z_nagp_max; m = 1;
while Z_nagp(m) > 0
    I_2(m) = U_2_XX / (sqrt(3) * (Z_k + Z_nagp(m)));
    U_2(m) = sqrt(3) * I_2(m) * Z_nagp(m);
```

```

m=m+1;
if m==70 %-для всей. m==24 - для вывода части вн. хар.
    break
end
Z_nagp(m) = 0.8*Z_nagp(m-1);
end
I_2_abs= abs(I_2); U_2_abs = abs(U_2);
plot(I_2_abs, U_2_abs);

```

Результат расчета представлен на рис. 1.12. Значение тока короткого замыкания на внешней характеристике совпал со значением, которое было рассчитано традиционным способом, т.е. через напряжение короткого замыкания. Внешняя характеристика трансформатора является нелинейной кривой.

Следует отметить, что эта особенность внешней характеристики в некоторых случаях не столь очевидна. Например, если рассматривать часть характеристики от нуля до номинального тока (рис. 1.13), то она практически линейна. Данная версия внешней характеристики позволяет более наглядно оценить изменение вторичного напряжения под действием нагрузки. Для её вывода в 10 строке программы необходимо переменную **m** сравнивать не с числом 70, а с 24, где **m** – количество выводимых точек.

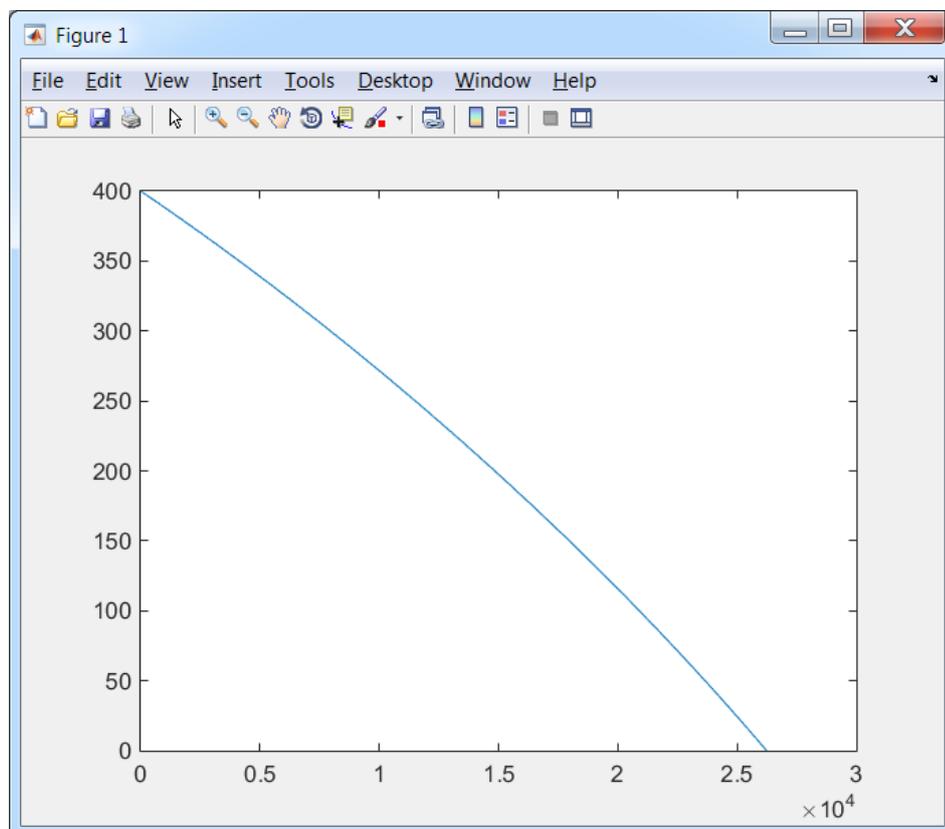


Рис. 1.12

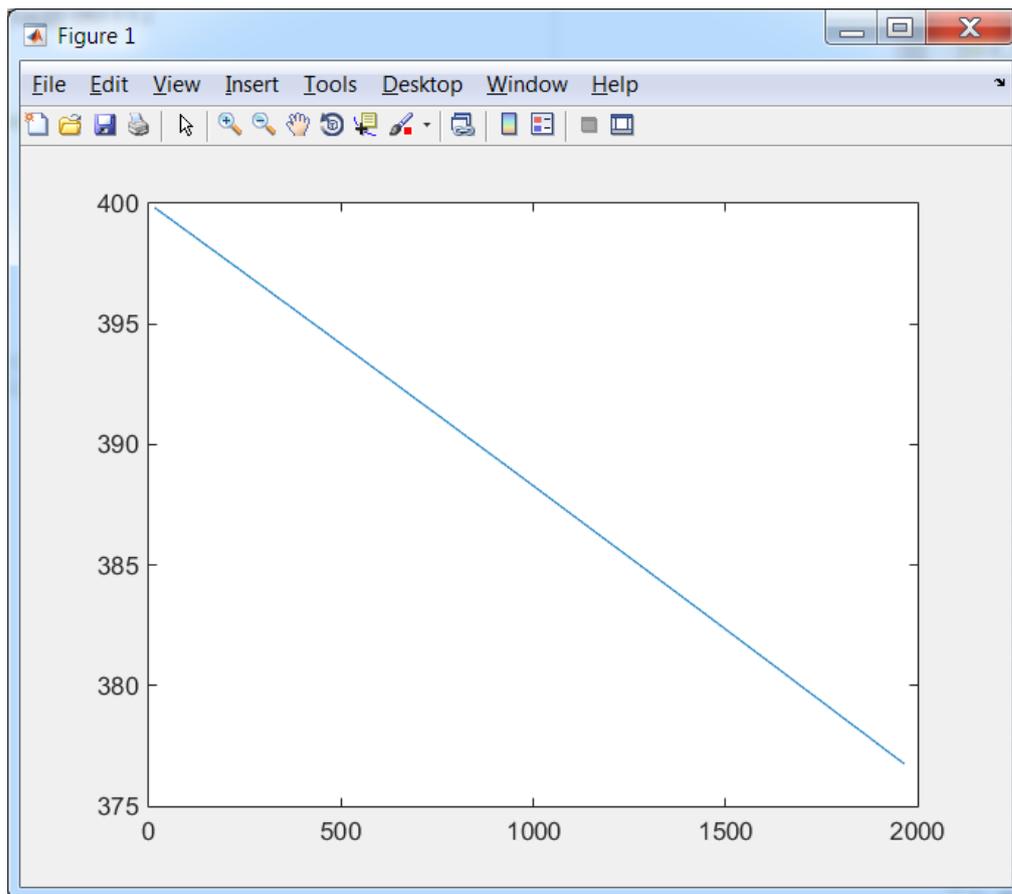


Рис. 1.13

В данном случае оператор прерывания **break** позволяет досрочно выйти из цикла по завершении 9 итераций.

Следует отметить, что оператор **break** не работает вне циклов **for** и **while**. В этом случае необходимо использовать оператор **return**, который обеспечивает возврат в вызывающую функцию.

Оператор **continue** передает управление в следующую итерацию цикла, пропуская операторы, которые записаны за ним, причем во вложенном цикле он передает управление на следующую итерацию основного цикла.

Для создания пауз в вычислениях или выводе на экран используется оператор **pause**.

При создании процедуры ветвления, часто прибегают к конструкции с переключателями типа **switch**. Такая конструкция позволяет решить задачу примера 1.2 более изящно.

Пример 1.4.

```
% example_1_4. Программа для определения цвета шины по её
% обозначению. Вызывает функцию faza (файл: faza.m).
for m = 1:10 % Организация цикла
    faza
end

% функция для программы example_1_4. Определяет цвет шины по
её обозначению.
function faza
% input - оператор ввода, 's'- для ввода строки символов,
faza = input(' Введите название шины (A; B; C; N): ', 's');
    switch faza
        case ('A')
            disp('Шина фазы "A" обозначается желтым цветом. ');
        case ('B')
            disp('Шина фазы "B" обозначается зеленым цветом. ');
        case ('C')
            disp('Шина фазы "C" обозначается красным цветом. ');
        case ('N')
            disp('Нулевой рабочий (нейтральный) проводник "N"
обозначается голубым цветом. ') % Указать одной строкой.
        otherwise
            disp('Уточнить раскладку клавиатуры, регистр или
букву. ') % В m-файле текст между '...' указать одной строкой.
            disp('Для выхода из цикла необходимо нажать
"Ctrl + C". ')
    end
end
end
```

В принципе, имеется возможность обращаться не к главной программе *example_1_4*, которая служит только для создания цикла, а непосредственно к функции **faza**. Однако, в этом случае придется каждый раз её запускать для получения ответа (рис. 1.14).

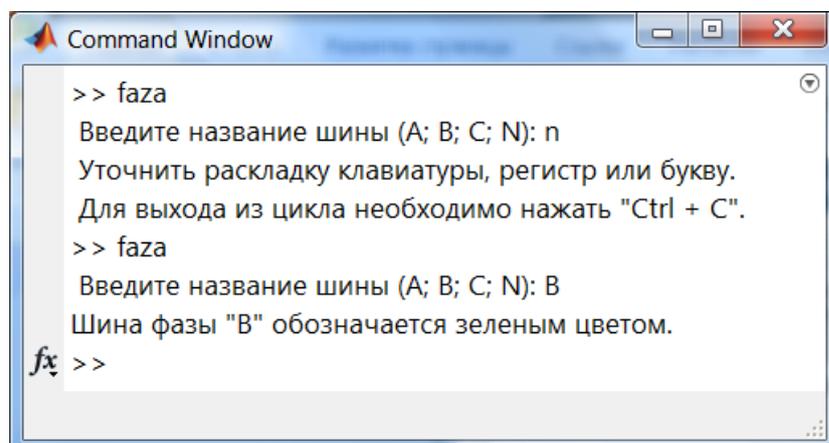


Рис. 1.14

1.1.8. Решение системы дифференциальных уравнений электротехнических устройств

1.1.8.1. Трансформатор

MATLAB позволяет рассчитать динамические режимы работы электротехнического устройства путем решения его системы дифференциальных уравнений.

Пример 1.5.

Рассчитать переходной процесс при трехфазном коротком замыкании трансформатора ТМ-1000/10 без учета намагничивающей ветви (Параметры из примера 1.3). Мгновенное значение напряжения изменяется по закону: $u_{2_XX} = \sqrt{2} \cdot U_{2\text{фаз_XX}} \sin(2\pi f_1 t)$, здесь $U_{2\text{фаз_XX}}$ – действующее значение фазного напряжения вторичной обмотки в режиме холостого хода, f_1 – частота напряжения.

Решение.

Дифференциальное уравнение напряжения одной фазы трансформатора имеет следующий вид [4]:

$$u_{2_XX} = r_k i_k + L_k \frac{di_k}{dt}. \quad (1.1)$$

Представим (1.1) в виде:

$$\frac{di_k}{dt} = \frac{\sqrt{2} \cdot U_{2\text{фаз_XX}} \sin(\omega t + \alpha_0) - r_k i_k}{L_k}. \quad (1.2)$$

```
% example_1_5. Программа расчета переходного процесса
% при трехфазном коротком замыкании трансформатора.
% Вызывает функцию funa(файл: funa.m) с правыми частями системы
% дифференциальных уравнений.
U_2faz_XX = 230.9; f_1=50; r_k = 0.00192; L_k = 2.734e-5;;
w=2*pi*f_1 % Угловая частота сети.
Fi_k = atan(L_k*w/r_k) % Угол в треугольнике короткого
замыкания
% Начальная фаза напряжения, при которой:
% alfa_0 = Fi_k; % 1) аperiodический ток отсутствует;
% alfa_0 = Fi_k + pi/2 % 2) аperiodический и ударный токи
максимальны.
tspan = [0:0.0005: 0.1]; % Время начала, шаг вывода данных,
время завершения.
y_0 = [0]; % Начальные значения переменных.
options = []; % Опции решателя - по умолчанию.
[T,Y]=ode45(@funa, tspan, y_0, options, U_2faz_XX, w, r_k,
L_k, alfa_0) ;
u = sqrt(2)*U_2faz_XX*sin(w*T + alfa_0);
```

```

plotyy(T,Y,T,u) % Построение графиков по двум осями, т.е.
Y=f(T) и u=f(T).
grid on % Сетка.

% Функция правых частей СДУ funa (файл: funa.m) для программы
example_1_5.
function dydt= funa(t, y, U_2faz_XX, w, r_k,L_k, alfa_0)%
Объявление матрицы dydt
dydt=zeros(1,1); % Инициализация заданной матрицы
dydt(1) = (sqrt(2)*U_2faz_XX*sin(w*t + alfa_0) -
r_k*y(1))/L_k; % Уравнение (1.2)
end

```

Результаты расчета тока короткого замыкания при благоприятной фазе напряжения ($\alpha_0 = \varphi_k = \arctg(\omega L_k / r_k) = 1.3509$) представлены на рис. 1.5, при неблагоприятной ($\alpha_0 = \varphi_k + \pi/2 = 1.3509 + \pi/2 = 2.9217$) – на рис. 1.6. В последнем случае, при $t = \pi / \omega = 0,01c$, ударный ток достигает максимального значения.

Следует отметить, что при появлении ломаной линии вместо синусоидальной кривой, необходимо уменьшить шаг вывода данных в `tspan`.

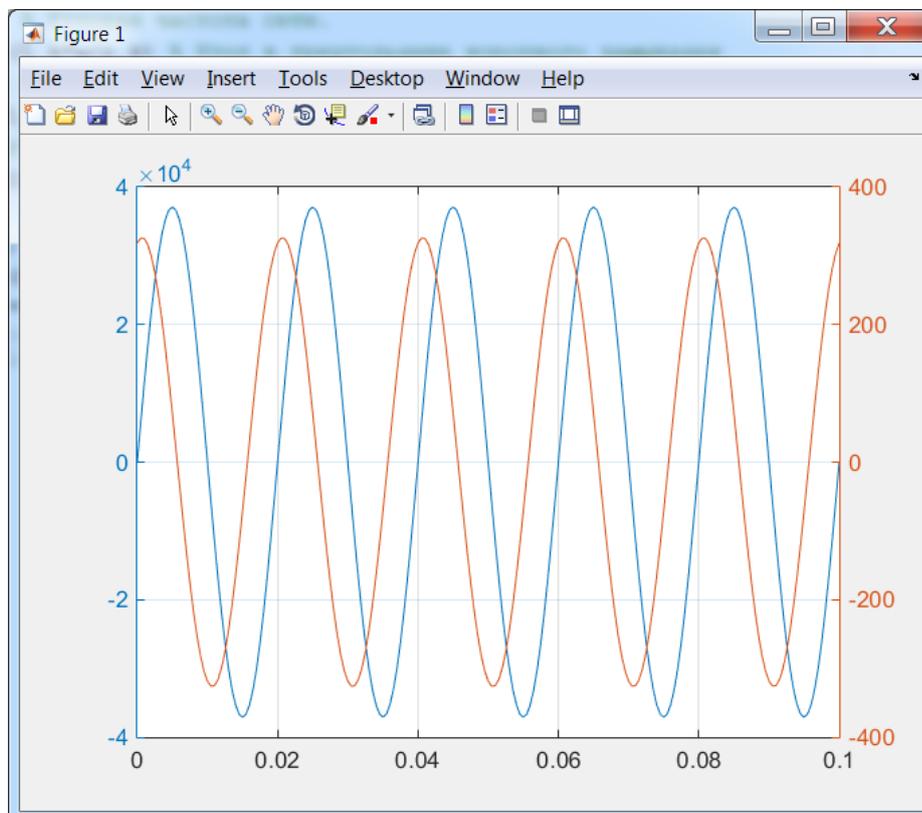


рис. 1.15

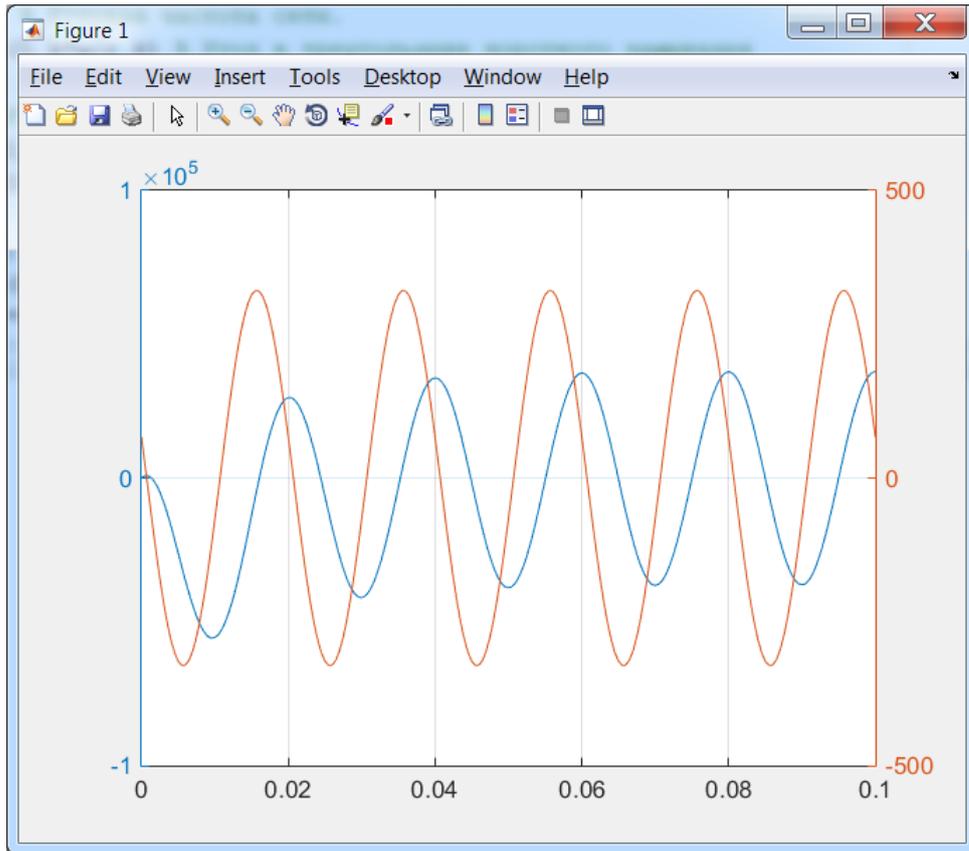


Рис. 1.16.

1.1.8.2. Двигатель постоянного тока

Наиболее простым электромеханическим преобразователем, с точки зрения математического описания, является коллекторный двигатель постоянного тока (ДПТ) с независимым или параллельным возбуждением. Такой двигатель описывается следующей системой дифференциальных уравнений [5]:

$$\left. \begin{aligned} u &= k\Phi\omega + ir + L\frac{di}{dt}; \\ M &= M_c + J\frac{d\omega}{dt}, \end{aligned} \right\} \quad (1.3)$$

где r , L – соответственно активное сопротивление и индуктивность якорной цепи; k – конструктивный коэффициент; Φ – магнитный поток двигателя; J – момент инерции двигателя и нагрузки. В свою очередь электромагнитный момент и ЭДС двигателя при постоянном потоке соответственно равны: $M = k\Phi i = ci$; $e_d = k\Phi\omega = c\omega$, здесь: i – ток якоря, $c = k\Phi$; ω – угловая скорость якоря (ротора) ДПТ.

Представим систему (1.3) в следующем виде:

$$\left. \begin{aligned} \frac{di}{dt} &= \frac{u - c\omega - ir}{L}; \\ \frac{d\omega}{dt} &= \frac{ci - M_c}{J}. \end{aligned} \right\} \quad (1.4)$$

Система дифференциальных уравнений позволяет исследовать динамические характеристики ДПТ. Рассмотрим следующий пример.

Пример 1.6.

Предположим, что технические данные ДПТ и его параметры в номинальном режиме имеют следующие значения: $P_H = 2 \text{ кВт}$ – мощность; $u_H = 110 \text{ В}$ – напряжение; $n_H = 3000 \text{ об/мин}$ – частота вращения; $\eta_H = 0,785$ – КПД; $r = 0,336 \text{ Ом}$ – сопротивление цепи якоря двигателя; $L = 6,6 \text{ мГн}$ – индуктивность якоря двигателя; $r_B = 265 \text{ Ом}$ – активное сопротивление обмотки возбуждения; $J_D = 0,011 \text{ кг} \cdot \text{м}^2$ – момент инерции якоря двигателя; $k_J = 1,25$ – коэффициент увеличения момента инерции за счет нагрузки, т.е. $J = k_J J_D = 0,01375 \text{ кг} \cdot \text{м}^2$.

Рассчитать динамические режимы ДПТ при пуске и набросе номинальной нагрузки. Построить зависимости $\omega = f(t)$ и $i = f(t)$.

Решение.

Рассчитаем коэффициент, который входит в систему уравнений (1.4):

$$c = k\Phi = \frac{u_H - i_H r}{\omega_H} = \frac{110 - 23,16 \cdot 0,336}{314,157} = 0,325 \text{ В} \cdot \text{с}, \quad (1.5)$$

где: $i_H = P_H / (u_H \eta_H) = 2000 / (110 \cdot 0,785) = 23,16 \text{ А}$ – номинальный ток; $\omega_H = (2\pi n_H) / 60 = (2\pi 3000) / 60 = 314,159 \text{ рад/с}$ – номинальная угловая скорость вращения двигателя.

Момент нагрузки примем равным номинальному моменту двигателя: $M_c = M = c \cdot i_H = 0,325 \cdot 23,16 = 7,527 \text{ Н} \cdot \text{м}$.

На основе (1.4) разработана программа расчета (m-файл), текст которой представлен ниже.

```
% example_1_6. Программа расчета (m-файл) переходного процесса
% ДПТ независимого (параллельного) возбуждения
% Вызывает функцию fun_DPT (файл: fun_DPT.m) с правыми частями
% системы дифференциальных уравнений.
u=110; r=0.336; L=6.6e-3; J=0.01375; M_c_nabr=7.527; c=0.325;
t_nabr = 0.5 ; % Время наброса номинальной нагрузки.
tspan = [0:0.0005: 0.8]; % Время начала, шаг вывода данных,
% время завершения.
y_0 = [0; 0]; % Начальные значения переменных.
```

```

options =[]; % Опции решателя - по умолчанию.
[T,Y]=ode45(@fun_DPT, tspan, y_0, options, u, r, L, J,
M_c_nabr, c, t_nabr) ;
plot(T,Y(:,1),T,Y(:,2)) % Построение графиков по двум осям,
% т.е. i=f(T) и w=f(T).

grid on % Сетка.

% Функция правых частей СДУ fun_DPT (файл: fun_DPT.m) для
% программы example_1_6.
function dydt= fun_DPT(t, y, u, R, L, J, M_c_nabr, c, t_nabr)
% Объявление матрицы dydt
dydt=zeros(2,1); % Инициализация заданной матрицы
if t<t_nabr % наброс нагрузки при t=t_nabr
M_c=0;
else
M_c=M_c_nabr;
end
dydt(1) = (u - c*y(2) - R*y(1))/L; % СДУ (1.4)
dydt(2) = (c*y(1) - M_c)/J;
end

```

Результаты моделирования прямого пуска ДПТ с последующим набросом номинальной нагрузки представлены на рис. 1.17. Ось тока – слева, ось частоты вращения – справа.

Следует отметить, что данная программа так же позволяет исследовать влияние параметров ДПТ, напряжения и величины статического момента нагрузки на динамические режимы.

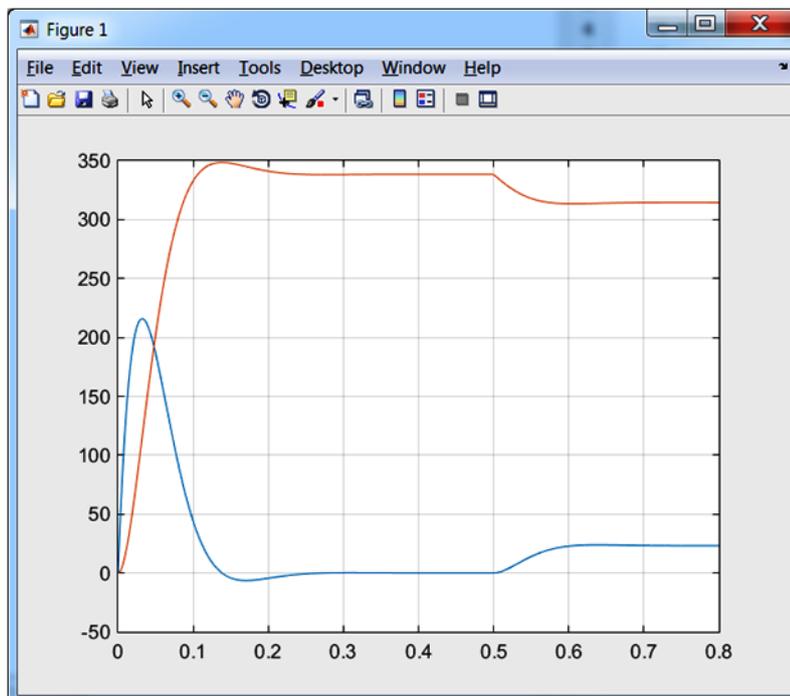


Рис. 1.17

Кривые, представленные на рис. 1.17, позволяют сделать очевидные выводы: при прямом пуске ДПТ на начальном этапе возникает пусковой ток, который в 9,3 раза больше номинального; на заключительном этапе разгона двигатель переходит в режим рекуперации; после разгона – работает в режиме идеального холостого хода (ток равен нулю); наброс номинальной нагрузки приводит к снижению частоты вращения ротора и увеличению тока, при этом в установившемся режиме они принимают номинальные значения.

1.1.8.3. Асинхронный двигатель с короткозамкнутым ротором

Поскольку основным потребителем электроэнергии на промышленных предприятиях остаются асинхронные машины, то исследование их динамических режимов представляет определенный интерес.

При реализации программы расчета воспользуемся известной системой дифференциальных уравнений асинхронной машины [5]:

$$\left. \begin{aligned} u_{1\alpha} &= r_1 i_{1\alpha} + \frac{d\Psi_{1\alpha}}{dt}; & u_{1\beta} &= r_1 i_{1\beta} + \frac{d\Psi_{1\beta}}{dt}; \\ 0 &= r_2 i_{2\alpha} + \frac{d\Psi_{2\alpha}}{dt} + \omega_p \Psi_{2\beta}; & 0 &= r_2 i_{2\beta} + \frac{d\Psi_{2\beta}}{dt} - \omega_p \Psi_{2\alpha}; \\ \frac{di_\alpha}{dt} &= \frac{u_{1\alpha} - ri_\alpha}{L}; & \frac{di_\beta}{dt} &= \frac{u_{1\beta} - ri_\beta}{L}; \\ M &= \frac{mpM}{2(L_1 L_2 - M^2)} (\Psi_{1\beta} \Psi_{2\alpha} - \Psi_{1\alpha} \Psi_{2\beta}); \\ \frac{d\omega_p}{dt} &= \frac{p}{J} (M - M_c), \end{aligned} \right\} \quad (1.5)$$

где $u_{1\alpha} = U_m \cos(\omega t)$, $u_{1\beta} = U_m \sin(\omega t)$ – напряжения на обмотке статора по осям α , β ; $i_{1\alpha}$, $i_{1\beta}$ – токи статора; $i_{2\alpha}$, $i_{2\beta}$ – токи ротора; i_α , i_β – токи сети; $i_{c\alpha}$, $i_{c\beta}$ – токи конденсаторов; $\Psi_{1\alpha}$, $\Psi_{1\beta}$, $\Psi_{2\alpha}$, $\Psi_{2\beta}$ – потокосцепления статора и ротора; r_1 , r_2 – активные сопротивления статора, ротора и сети; ω , ω_p – частота сети и частота вращения ротора (эл. рад/с); L_1 , L_2 – полные индуктивности обмоток статора и ротора; L_m – взаимная индуктивность между обмотками; M , M_c – электромагнитный момент АД и статический момент нагрузки; m – число фаз; p – число пар полюсов; J – момент инерции двигателя и механизма.

Потокосцепления обмоток в (1.5):

$$\left. \begin{aligned} \Psi_{1\alpha} &= L_1 i_{1\alpha} + M i_{2\alpha}; \Psi_{1\beta} = L_1 i_{1\beta} + M i_{2\beta}; \\ \Psi_{2\alpha} &= L_2 i_{2\alpha} + M i_{1\alpha}; \Psi_{2\beta} = L_2 i_{2\beta} + M i_{1\beta}. \end{aligned} \right\} \quad (1.6)$$

Пример 1.7.

В качестве примера, рассчитаем динамический режим пуска АД, который имеет следующие параметры: $P_2 = 110 \text{ кВт}$; $U_n = 220 \text{ В}$; $r_1 = 0,02155 \text{ Ом}$; $r_2 = 0,01231 \text{ Ом}$; $L_1 = L_2 = 0,010646 \text{ Гн}$; $M = 0,01038 \text{ Гн}$; $J_{\text{об}} = 2,3 \text{ кгм}^2$; $p = 2$ [5]. Параметры обмотки ротора приведены к обмотке статора.

Решение.

Для расчета динамических режимов АД воспользуемся следующей программой (m-файлом):

```
% example_1_7. Программа расчета (m-файл) динамических режимов
% работы АД с короткозамкнутым ротором
% Вызывает функцию fun_AD (файл: fun_AD.m) с правыми частями
% системы дифференциальных уравнений и параметрами АД.
global U r_1 L_1 r_2 L_2 M p J w_1 M_c nabr t_nabr;
U=220; % Действующее значение напряжения на статоре.
r_1=0.02155; % Активное сопротивление статора.
L_1=0.010606; % Полная индуктивность обм. статора.
r_2=0.01231; % Активное сопротивление ротора.
L_2=L_1; % Полная индуктивность обм. ротора
M=0.01038; % Взаимная индуктивность.
p=2; % Число пар полюсов.
J=2.3; % Момент инерции.
f_1 = 50; % Частота сети.
w_1 = 2*3.14159*f_1; % Угловая частота сети.
M_c_nabr = 706.4; % Статический момент нагрузки, равный
% номинальному моменту АД.
t_nabr = 1; % Время наброса статического момента
tspan = 0:0.0005: 1.4; % Время начала расчета. Шаг вывода.
% Время завершения.
y_0 = [0; 0; 0; 0; 0]; % Начальные значения переменных.
[T,Y]=ode45(@fun_AD, tspan, y_0);
Y_out = [];
Y_out(:,1) = (Y(:,1).*L_2 - Y(:,3).*M)/(L_1*L_2 - M*M); % ток
статора по оси альфа
Y_out(:,2) = (Y(:,2).*L_2 - Y(:,4).*M)/(L_1*L_2 - M*M); % ток
статора по оси бета
Y_out(:,3) = (Y(:,3).*L_1 - Y(:,1).*M)/(L_1*L_2 - M*M); % ток
ротора по оси альфа
Y_out(:,4) = (Y(:,4).*L_1 - Y(:,2).*M)/(L_1*L_2 - M*M); % ток
ротора по оси бета
Y_out(:,5) = Y(:,5)/p; % Угловая скорость ротора.
Y_out(:,6) = 1.5*p*M*(Y(:,1).*Y(:,4) - Y(:,2).*Y(:,3))/(L_1*L_2 -
M*M); % Электромагнитный момент АД.
```

```

% Построение графиков скорости ротора и электромагнитного
% момента АД.
plotyy(T,Y_out(:,5),T,Y_out(:,6))
grid on % Сетка.

% функция правых частей СДУ fun_AD (файл: fun_AD.m) для
программы example_1_7.
% Объявление матрицы dydt
function dydt= fun_AD(t, y)
global U r_1 L_1 r_2 L_2 M p J w_1 M_c_nabp t_nabp;
dydt=zeros(5,1); % Инициализация заданной матрицы
if t<t_nabp % наброс нагрузки при t=t_nabp
    M_c=0;
else
    M_c=M_c_nabp;
end
dydt(1) = - ((y(1)*L_2 - y(3)*M)/(L_1*L_2 - M*M))*r_1 +
sqrt(2)*U*sin(w_1*t);
dydt(2) = - ((y(2)*L_2 - y(4)*M)/(L_1*L_2 - M*M))*r_1 +
sqrt(2)*U*cos(w_1*t);
dydt(3) = - ((y(3)*L_1 - y(1)*M)/(L_1*L_2 - M*M))*r_2 +
y(5)*y(4);
dydt(4) = - ((y(4)*L_1 - y(2)*M)/(L_1*L_2 - M*M))*r_2 -
y(5)*y(3);
dydt(5) = p*(1.5*p*M*(y(1)*y(4) - y(2)*y(3)))/(L_1*L_2 - M*M) -
M_c/J;
end

```

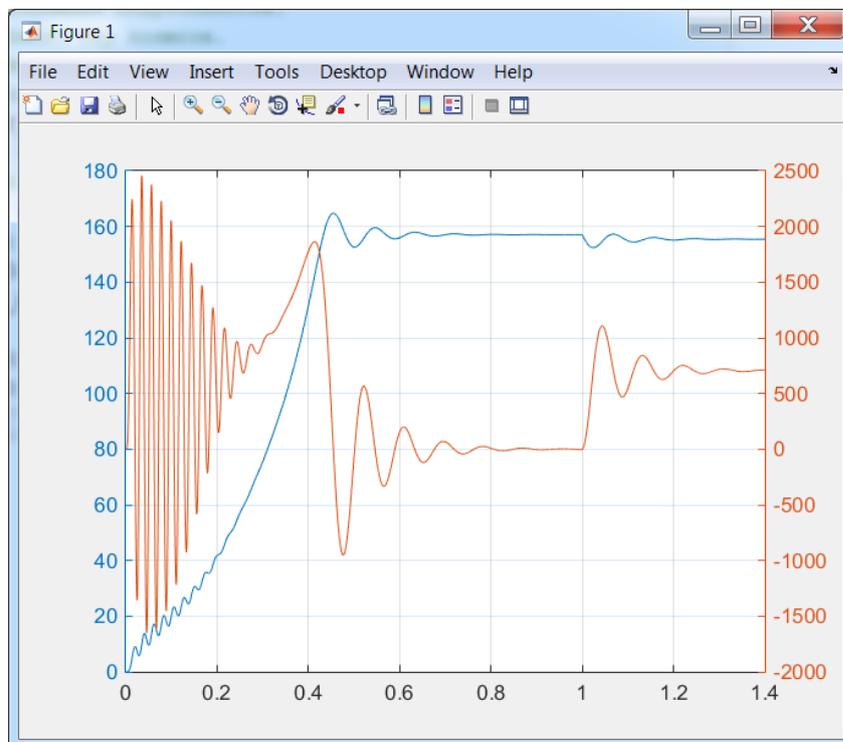


Рис. 1. 18

Результаты расчета динамических режимов работы асинхронного двигателя представлены на рис. 1.18. Двигатель разгоняется при отсутствии нагрузки на валу. В момент времени $t = 1с$ на вал АД подается статический момент нагрузки, который в данном случае равен номинальному моменту двигателя, т.е., $M_C = M_{ном} = (P_{2ном} / 60) / (2\pi n_{ном}) = 706,4 Нм$.. Как при пуске, так и при набросе нагрузки наблюдается затухающий колебательный процесс. До номинальной частоты вращения данный АД без нагрузки на валу разгоняется чуть меньше, чем за 0,6 сек.

При пуске АД с 45%-ым от номинального значения моментом, время пуска возрастает (рис. 1.19). К аналогичному результату приводит увеличение момента инерции и снижение напряжения.

Следует отметить, что по мере усложнения программы расчета возрастают и проблемы, связанные с её отладкой. Существенно облегчить поиск ошибок позволяет отладчик m-файлов, который гармонично взаимодействует с редактором файлов.

Рассмотрим работу отладчика на примере последней программы. Предположим, что нам необходимо проверить, в какой момент времени происходит операция присваивания $m_c = m_c_nabp$ в функции fun_AD .

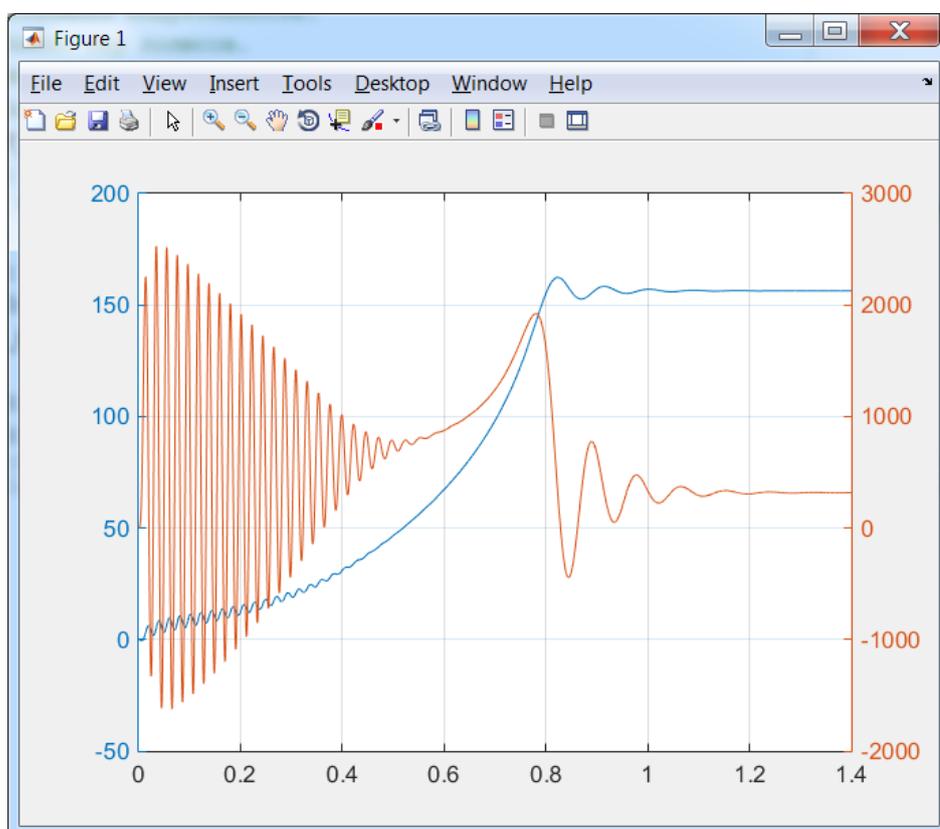


Рис. 1.19

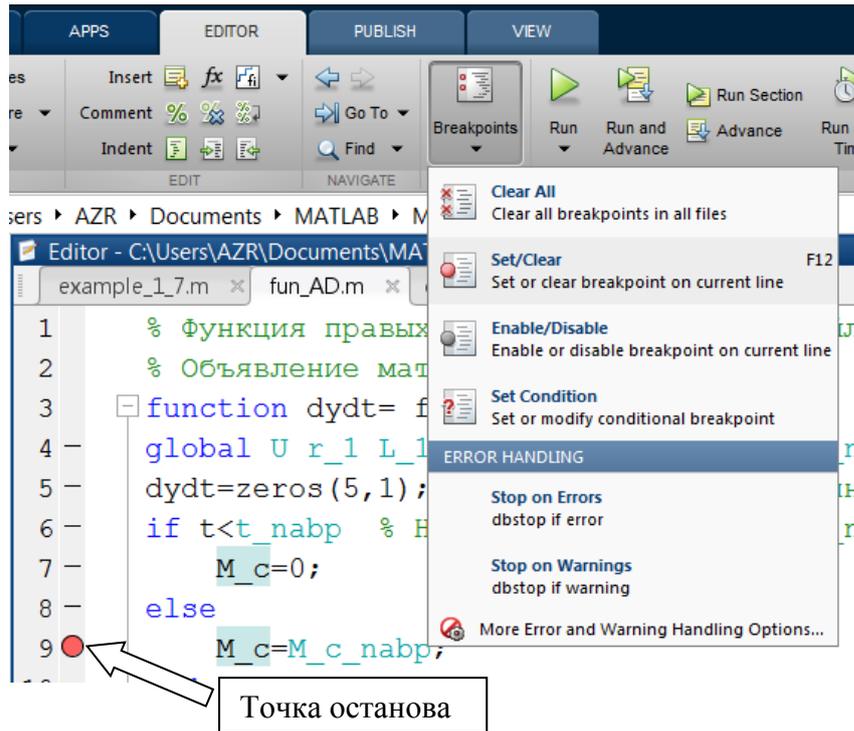


Рис. 1.20

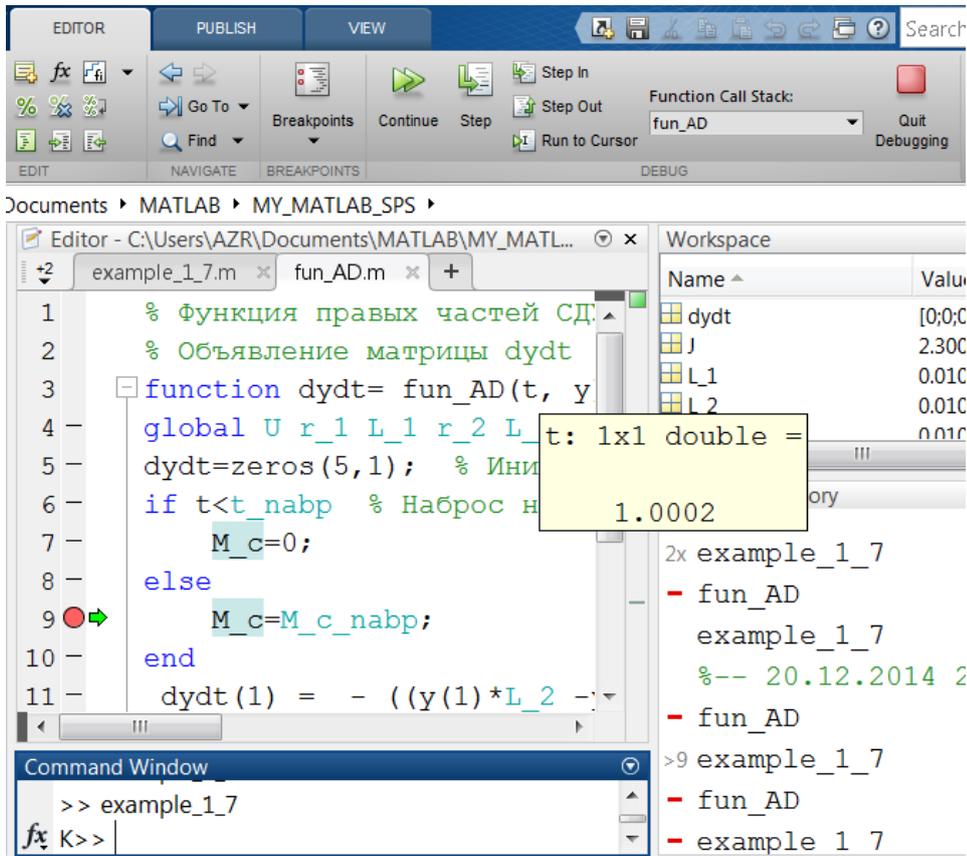


Рис. 1.21

Для этого установим в этой строке точку останова путем щелчка мышью в колонке останова (рис. 1.20). Альтернативные варианты: использовать кнопку на панели инструментов **Breakpoints** → **Set/Clear**; использовать клавишу **F12**. Повторный щелчок удаляет точку останова. После этого переходим в окно программы *example_1_7* и запускаем её, нажав на кнопку **Run**, расположенную на панели инструментов. После этого программа достигает точки останова и **MATLAB** переходит в режим отладки (рис. 1.21), при этом символ приглашения **>>** в командном окне меняется на **K>>**, а рядом с точкой останова появляется зеленая стрелка. Если необходимо определить содержимое переменной, то достаточно подвести к нему стрелку. После этого появляется окно с информацией о переменной, в данном случае это переменная времени *t*.

В режиме отладки имеется возможность пошагового выполнения программы. Для этого достаточно воспользоваться кнопками на панели инструментов или клавишей **F10**. При этом вы можете просматривать содержимое переменных и выражений. Для выхода из режима отладки достаточно нажать красную кнопку **Quit Debugging**.

Описанные выше подходы к программированию и визуализации результатов в настоящее время доступны и в других вычислительных системах. Однако, одним из достоинств, которое позволило **MATLAB** занять лидирующее положение в области современного математического и научно-технического программного обеспечения, стали многочисленные **расширения для технических вычислений**. Эти расширения позволяют создать модель всего устройства в целом, несмотря на различия в их физической природе. Например, если речь идет об автомобиле, то одновременно рассчитать динамические характеристики его взаимосвязанных энергетических, механических, электрических и иных узлов, т.е. смоделировать поведение всего автомобиля при разгоне, торможении, повороте и т.д. Это позволяет на этапе проектирования, а не на этапе создания опытного образца, обнаружить значительную часть ошибок и устранить их. Поскольку стоимость опытного образца не сопоставима со стоимостью его модели, то такая "работа над ошибками" позволяет сэкономить значительные средства при создании нового технического устройства [6].

Что касается электроэнергетики, то использование **MATLAB** и её расширений позволяет: настраивать устройства релейной защиты на модели энергосистемы или её участка, а не на самом объекте; моделировать устойчивость и иные режимы работы энергосистемы с учетом электрической, механической и иных составляющих; получать и обрабатывать информацию с устройств контроля и учета [3].

В следующем разделе рассмотрим основы работы с **Simulink** – основным расширением **MATLAB**, которое позволяет осуществить блочное имитационное моделирование различных систем и устройств с применением визуально-ориентированного программирования [2 , 3 , 7 , 8].

1.2. Simulink

1.2.1. Система моделирования Simulink

Основным достоинством расширения **Simulink** является простота и наглядность его использования при моделировании различных устройств и систем, в том числе и электротехнических. В основном это связано с тем, что вы не имеете дел с написанием строк программы, их редактированием и отладкой, как это имеет место в **MATLAB** или любом ином языке высокого уровня. В **Simulink** используется совершенно иной подход – визуально-ориентированный. При таком моделировании используются готовые блоки, которые необходимо с помощью мыши перенести из библиотеки в окно документа **Simulink**, соединить линиями входы и выходы этих блоков. В результате получаем S-модель, т.е. **Simulink** модель, которую запускаем простым нажатием кнопки **Run**.

1.2.2. Состав библиотеки Simulink

Библиотека **Simulink** представляет собой набор различных визуальных блоков. Для доступа к ним необходимо нажать кнопку **Simulink Library** на панели инструментов **MATLAB** (рис. 1.1). При этом появляется окно браузера (обозревателя, навигатора, программы просмотра) библиотеки, представленное на рис. 1.22.

Окно содержит следующие элементы:

- 1) Заголовок, с названием окна – **Simulink Library Browser**.
- 2) Панель инструментов, с ярлыками наиболее часто используемых команд.
- 3) Окно комментария для вывода поясняющего сообщения о выбранном блоке.
- 4) Список разделов библиотеки, реализованный в виде дерева.
- 5) Окно содержимого раздела библиотеки (список вложенных разделов библиотеки или блоков).

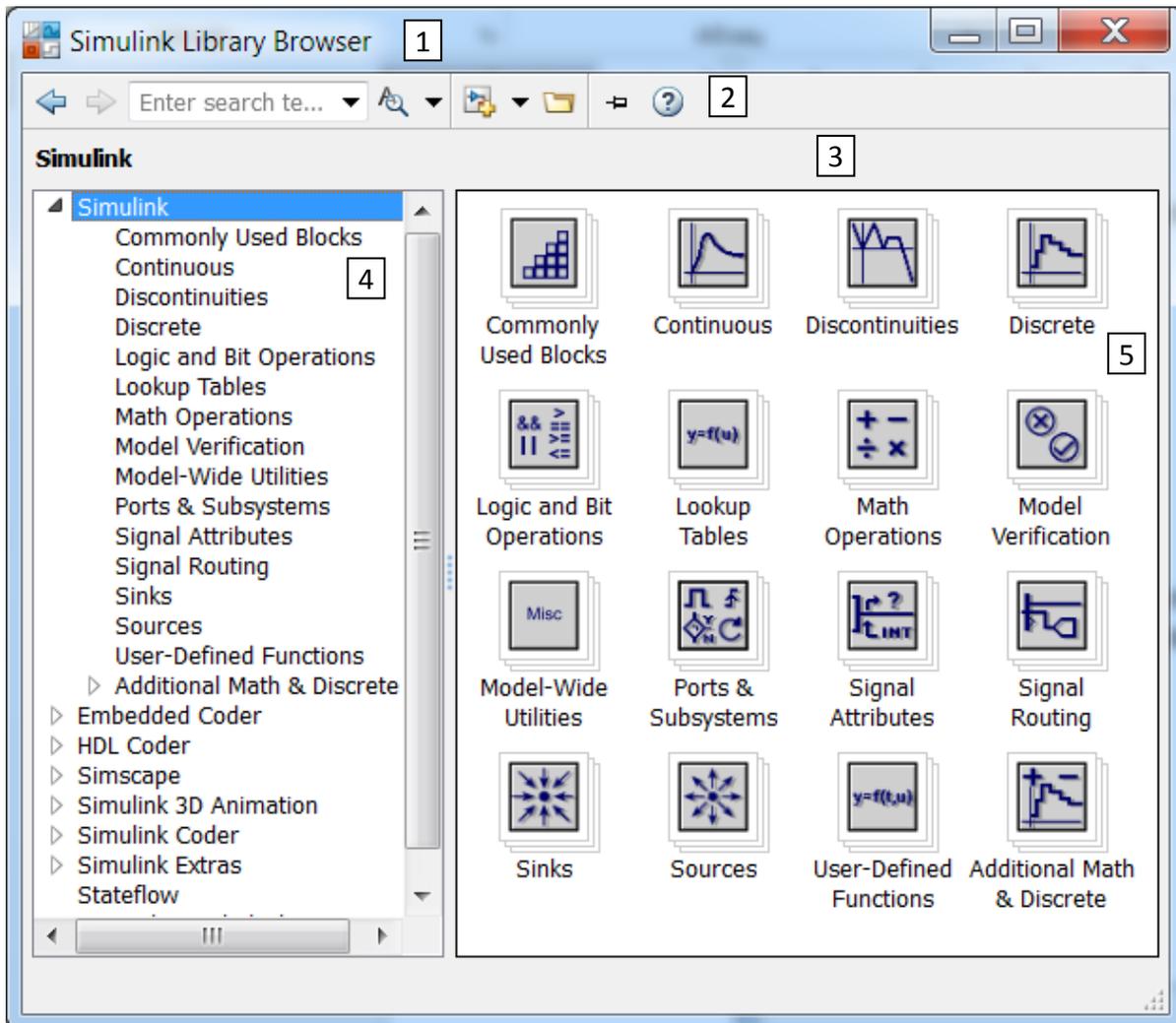
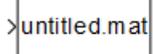


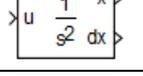
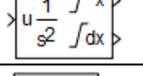
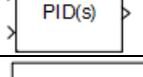
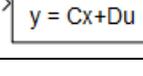
Рис. 1.22.

Ниже представлены некоторые разделы и блоки библиотеки **Simulink**. Полный перечень блоков доступен в **Help**. Для его получения достаточно два раза щелкнуть по любому блоку и в открытом окне параметров блока нажать кнопку **Help**.

Sources – источники сигналов			
1		Band-Limited White Noise	Генератор белого шума.
2		Chirp Signal	Генератор линейно-изменяющейся частоты.
3		Clock	Источник времени.
4		Constant	Источник постоянного сигнала.

5		Counter Free-Running	Источник сигнала типа «счетчик».
6		Counter Limited	Источник типа «счетчик с ограничением».
7		Digital Clock	Дискретный источник времени.
8		Enumerated Constant	Перечислимая константа.
9		From File	Блок считывания данных из файла.
10		From Workspace	Блок считывания данных из рабочей области MATLAB.
11		Ground	Формирователь сигнала нулевого уровня.
12		In1	Входной порт.
13		Pulse Generator	Источник импульсного сигнала.
14		Ramp	Источник линейно-изменяющегося воздействия.
15		Random Number	Источник случайного сигнала с нормальным распределением.
16		Repeating Sequence	Источник периодического сигнала с интерполяцией.
17		Repeating Sequence Interpolated	Источник периодического сигнала.
18		Repeating Sequence Stair	Источник ступенчатого периодического сигнала.
19		Signal Builder	Конструктор сигналов.
20		Signal Generator	Генератор сигналов.
21		Sine Wave	Источник синусоидального сигнала.
22		Step	Генератор ступенчатого сигнала.
23		Uniform Random Number	Источник случайного сигнала с равномерным распределением.

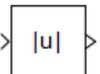
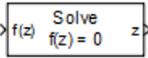
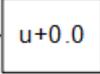
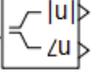
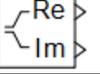
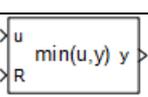
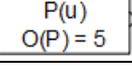
Sinks – приемники сигналов		
1	 Display	Цифровой дисплей.
2	 Floating Scope	«Плавающий» осциллограф.
3	 Out1	Выходной порт.
4	 Scope	Осциллограф.
5	 Stop Simulation	Блок остановки моделирования.
6	 Terminator	Концевой приемник.
7	 To File	Блок записи в файл.
8	 To Workspace	Блок записи в рабочую область MATLAB.
9	 XY Graph	Графопостроитель.

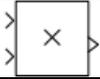
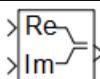
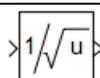
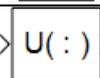
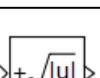
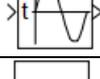
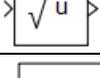
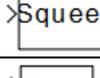
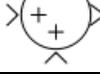
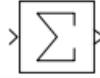
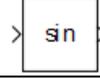
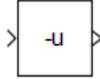
Continuous – блоки непрерывных моделей		
1	 Derivative	Блок вычисления производной.
2	 Integrator	Интегратор.
3	 Integrator Limited	Интегратор с ограничением.
4	 Integrator, Second-Order	Интегратор второго порядка.
5	 Integrator, Second-Ord...	Интегратор второго порядка с ограничением.
6	 PID Controller	ПИД-регулятор (контроллер).
7	 PID Controller (2DOF)	ПИД-регуляторы с двумя степенями свободы (<i>two-degree-of-freedom</i>).
8	 State-Space	Объект пространства состояний.

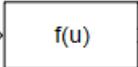
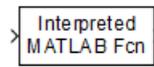
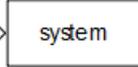
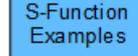
9		Transfer Fcn	Передаточная функция.
10		Transport Delay	Блок фиксированной задержки сигнала.
11		Variable Time Delay	Блок управляемой задержки времени.
12		Variable Transport D...	Блок управляемой задержки сигнала.
13		Zero-Pole	Передаточная функция «нули-полюса».

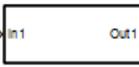
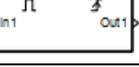
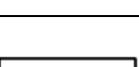
Discontinuities – нелинейные блоки			
1		Backlash	Люфт.
2		Coulomb & Viscous Fric...	Блок сухого и вязкого трения.
3		Dead Zone	Зона нечувствительности.
4		Dead Zone Dynamic	Управляемая зона нечувствительности.
5		Hit Crossing	Блок определения момента пересечения порогового значения.
6		Quantizer	Квантователь.
7		Rate Limiter	Управляемый блок ограничения сигнала.
8		Rate Limiter Dynamic	Управляемый блок ограничения скорости изменения сигнала.
9		Relay	Реле.
10		Saturation	Ограничитель.
11		Saturation Dynamic	Управляемый ограничитель.
12		Wrap To Zero	Блок, выполняющий сброс сигнала до нулевого уровня.

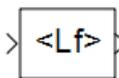
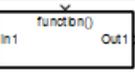
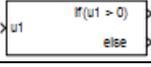
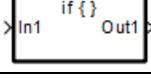
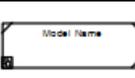
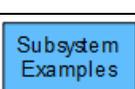
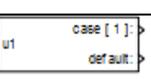
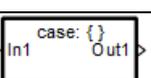
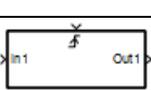
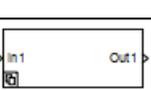
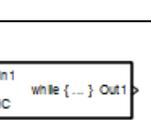
Math Operations – блоки математических операций

1		Abs	Блок вычисления модуля.
2		Add	Сумматор (пиктограмма в виде прямоугольника).
3		Algebraic Constraint	Блок алгебраического контура.
4		Assignment	Блок присвоения элементам массива новых значений.
5		Bias	Смещение.
6		Complex to Magnitude-...	Блок вычисления модуля и (или) аргумента комплексного числа.
7		Complex to Real-Imag	Блок вычисления действительной и (или) мнимой части комплексного числа.
8		Divide	Блок деления.
9		Dot Product	Блок скалярного умножения векторов.
10		Find Nonzero Elements	Блок поиска ненулевых элементов.
11		Gain	Усилитель.
12		Magnitude-Angle to Co...	Блок расчета комплексного числа по его модулю и аргументу.
13		Math Function	Математическая функция.
14		Matrix Concatenate	Блок объединения сигналов в матрицу.
15		MinMax	Блок вычисления максимального или минимального значения с возможностью сброса.
16		MinMax Running Reset...	Блок вычисления максимального или минимального значения.
17		Permute Dimensions	Блок транспонирования.
18		Polynomial	Степенной многочлен.

19		Product	Умножитель.
20		Product of Elements	Блок вычисления произведения элементов вектора.
21		Real-Imag to Complex	Блок расчета комплексного числа по его действительной и мнимой части.
22		Reciprocal Sqrt	Блок вычисления единицы деленной на квадратный корень.
23		Reshape	Преобразователь размерности сигнала.
24		Rounding Function	Блок округления числа.
25		Sign	Блок определения знака числа. При 0 на входе 0 на выходе.
26		Signed Sqrt	Квадратный корень со знаком. Например, при $u = -2$ на выходе получим "- 1.4142".
27		Sine Wave Function	Синусоидальная функция.
28		Slider Gain	Ползунковый регулятор.
29		Sqrt	Квадратный корень
30		Squeeze	Блок сжатия.
31		Subtract	Блок вычисления разности.
32		Sum	Сумматор (круглая пиктограмма).
33		Sum of Elements	Блок вычисления суммы элементов вектора.
34		Trigonometric Function	Тригонометрическая функция.
35		Unary Minus	Унарный минус.
36		Vector Concatenate	Блок объединения векторов.
37		Weighted Sample Tim...	Блок поддержки вычислений, использующих шаг дискретизации.

User-Defined Functions – функции, задаваемые пользователем		
1	 Fcn	Блок задания функции.
2	 Interpreted MATLAB Fu...	Блок ввода функций или выражений MATLAB.
3	 Level-2 MATLAB S-...	Блок задания S-функции второго уровня.
4	 MATLAB Function	Блок создания MATLAB-функции для использования в Simulink.
5	 MATLAB System	Блок для использования системных объектов.
6	 S-Function	S-функция.
7	 S-Function Builder	Конструктор S-функции на языке C.
8	 S-Function Examples	Примеры S-функции.

Порты и подсистемы Ports & Subsystems		
1	 Atomic Subsystem	Неделимая подсистемы.
2	 CodeReuseS-ubsystem	Подсистема, имеющая общий код для всех ее экземпляров в модели.
3	 Configurable Subsystem	Блок конфигурации подсистем.
4	 Enable	Создание порта для управления подсистемой.
5	 Enabled Subsystem	Блок создания управляемой подсистемы, т.е. E-подсистемы.
6	 Enabled and Triggered S...	Управляемая уровнем и фронтом сигнала подсистема, т.е. ET-подсистема.
7	 For Each Subsystem	Позволяет многократно выполнять алгоритм на каждом элементе или подмассиве.
8	 For Iterator Subsystem	Блок создания подсистемы, выполняющей итерационные действия под управлением цикла типа for.

9		Function-Call Feedback L...	Прерывает обратную связь между блоками.
10		Function-Call Generator	Внешний блок управления подсистемой с запросом функции.
11		Function-Call Split	Обеспечивает разветвление сигнала.
12		Function-Call Subsystem	Блок создания подсистемы с запросом функции.
13		If	Блок условного оператора.
14		If Action Subsystem	Управляемая условием подсистема.
15		In1	Входной порт.
16		Model	Блок, обеспечивающий подключение файла к текущей модели.
17		Model Variants	Позволяет одну модель использовать как блок в другой модели.
18		Out1	Выходной порт.
19		Subsystem	Блок позволяет создать подсистему.
20		Subsystem Examples	Примеры подсистем.
21		Switch Case	Переключаемая подсистема.
22		Switch Case Action Subs...	Переключаемая активная подсистема.
23		Trigger	Создание триггерного (пускового) порта для запуска подсистемы.
24		Triggered Subsystem	Блок подсистемы с триггерной системой управления.
25		Variant Subsystem	Представляет собой подсистему с несколькими подсистемами.
26		While Iterator Subsystem	Блок создания подсистемы, выполняющей итерационные действия под управлением цикла типа While.

1.2.2. Создание, редактирование и запуск Simulink модели

Для создания **Simulink** модели нажимаем кнопку  **New model** панели инструментов окна браузера (программы просмотра) библиотеки **Simulink Library** (рис. 1.22). К аналогичному результату приводят действия с вкладки основного окна **MATLAB: HOME** → **New** → **Simulink Model**. При этом создается пустое окно с названием файла **untitled** (рис. 1.23).

Сохраняем эту модель в новой папке, например, *example_1_8*, под другим названием, например, *example_1_8*, с расширением *.slx или *.mdl. Для этого последовательно нажимаем на панели инструментов **File** → **Save As...** и т.д. Следует отметить, что использование кириллицы в названии файла, папки или во всём пути к файлу может стать причиной появления ошибки при сохранении модели в файле с расширением *.mdl (*Не совпадают кодовые страницы операционной системы и MATLAB*).

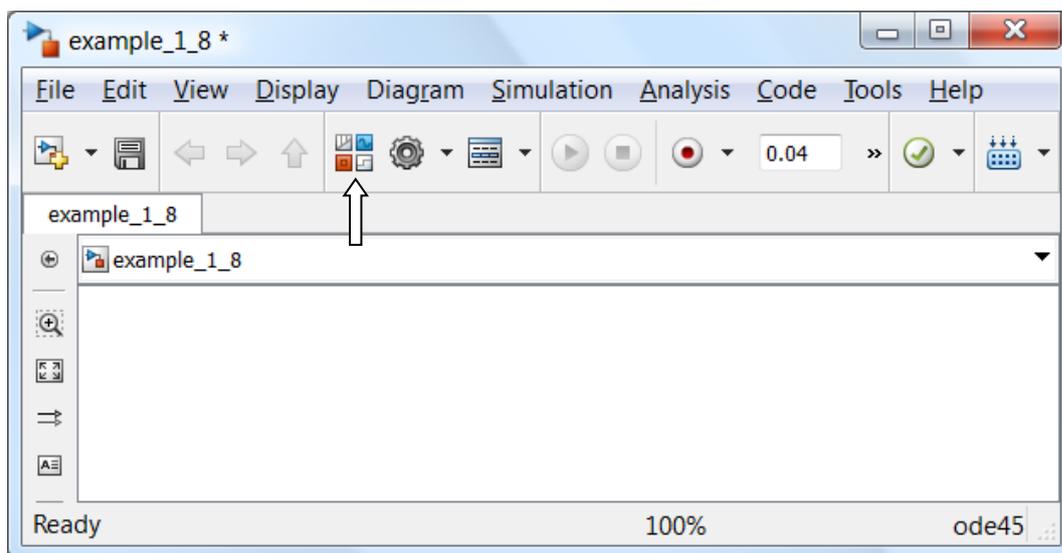


Рис. 1.23

Изучение возможностей **Simulink** начнем с простого примера.

Пример 1.8.

Предположим, что необходимо вывести на осциллограф синусоиду с действующим напряжением 230,9 В и частотой 50 Гц из *Примера 1.5*.

Для этого вызываем **Simulink Library Browser** путем нажатия в панели инструментов на кнопку  **Library Browser** (рис. 1.23). Для удобства окно библиотеки **Simulink Library Browser** располагаем слева от окна модели таким образом, чтобы они не перекрывали друг друга. В принципе, окно библиотеки можно закрепить поверх остальных открытых окон, нажав на кнопку  **Stay on top** (Остаться на переднем плане). В библиотеке **Simulink** переходим в раздел **Sources** (источники

сигналов), в котором наводим курсор на блок синусоидального сигнала **Sine Wave**. Затем нажимаем правую кнопку мыши, перетаскиваем этот блок к левому краю окна модели и отпускаем кнопку. Аналогично поступаем с блоком осциллографа **Scope** из раздела **Sinks** (приемники сигналов), но располагаем справа от первого блока. Теперь можно приступить к соединению выхода источника с входом приемника. Для этого подводим курсор к маленькому треугольнику "▷" на правой стороне блока **Sine Wave**. После того, как курсор попадет в область этого треугольника его вид изменится на +. Нажимаем на левую кнопку мыши и перетаскиваем красную пунктирную линию к входу блока **Sine Wave**, который изображен в виде ">". После изменения цвета и характера линии отпускаем левую кнопку. Соединение установлено (рис. 1.24).

Аналогичный результат можно получить, если удерживая клавишу **Ctrl**, вначале выделить первый блок, а после – второй. Для устранения линии связи достаточно её выделить и нажать **Delete**.

Следует отметить, что для удобного размещения блоков модели в окне **Simulink** можно воспользоваться кнопкой  на боковой панели инструментов или роликом мыши.

Далее два раза щелкаем левой кнопкой мыши по блоку **Sine Wave**. В раскрывшемся окне **Source Block Parameters: Sine Wave** задаем параметры синусоиды: $U_m = \sqrt{2}U = \sqrt{2} \cdot 220$; $\omega = 2\pi f = 2\pi 50$, и нажимаем на кнопку **OK** (рис. 1.25).

В маленьком окошке, расположенном в панели инструментов, изменяем время окончания расчета 10 с на 0,04 с.

Изменим ещё одну позицию, которая задана по умолчанию, – это максимальный шаг решателя. Для этого на панели инструментов нажимаем на кнопку .

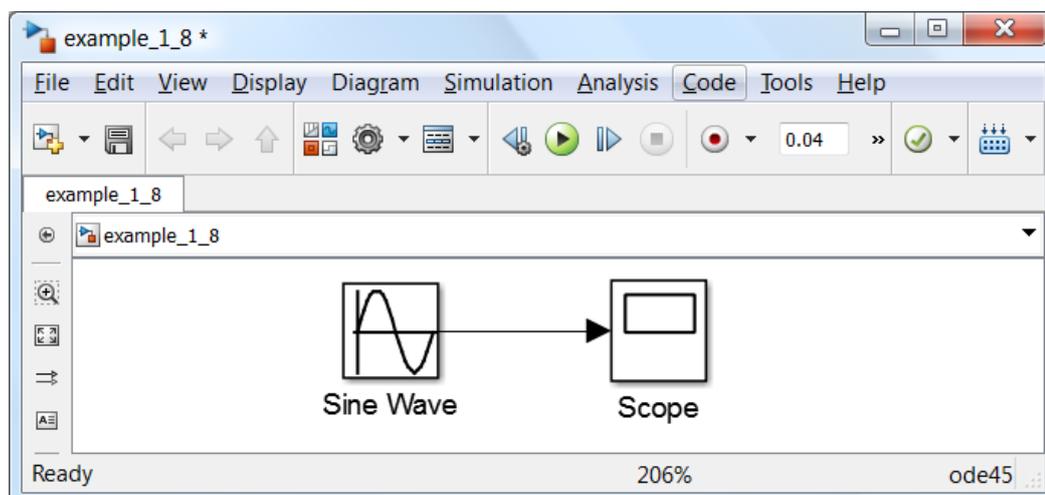


Рис. 1.24

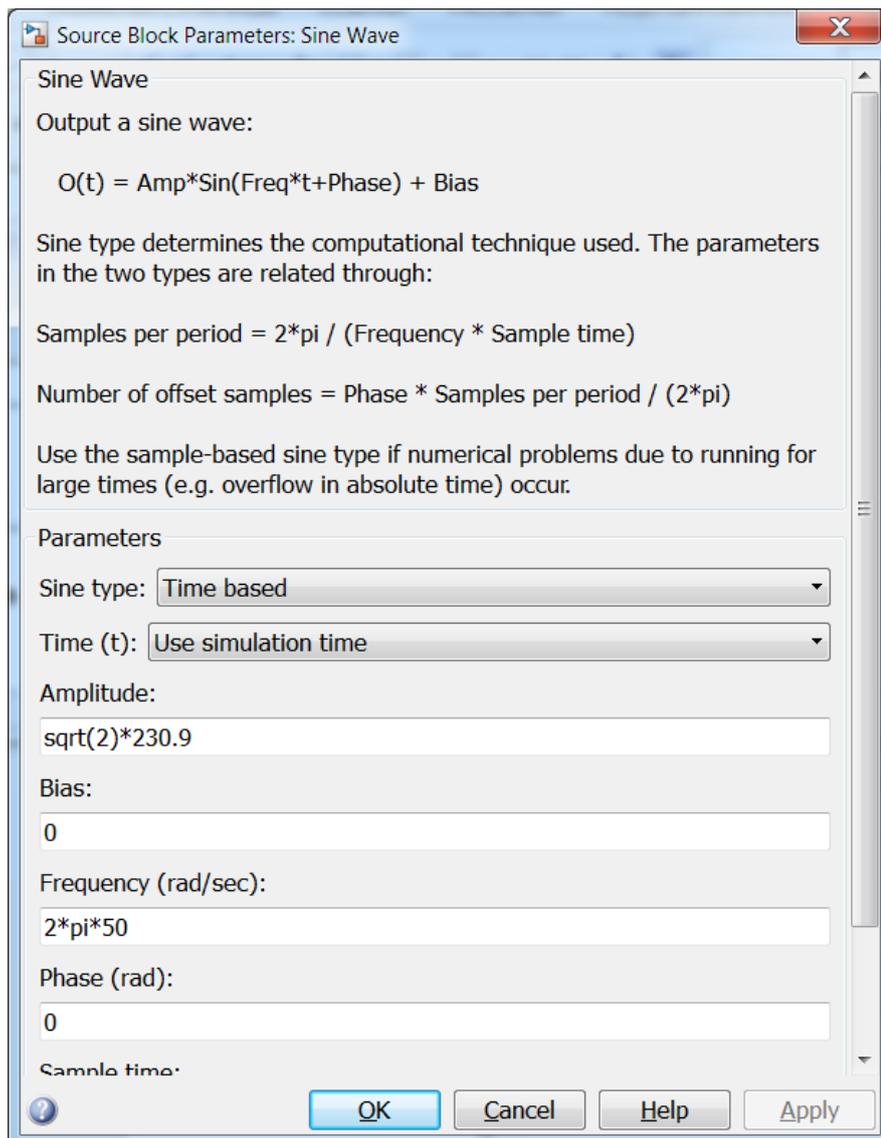


Рис. 1.25

В открывшемся окне **Model Configuration Parameters** находим окошко **Max step size** и вводим значение 0,0005 (рис. 1.26). Такое изменение позволит получить синусоидальную кривую вместо ломаной линии, поскольку возрастет количество точек, выводимых за один период.

Нажимаем кнопку  **Run**, расположенную на панели инструментов, и запускаем расчет. Результаты расчета в виде окна, представленного на рис. 1.27, раскрываем двойным щелчком мыши по блоку **Scope**. Если кривые выходят за границы осциллографа, то следует нажать кнопку  **Autoscale** в панели инструментов.

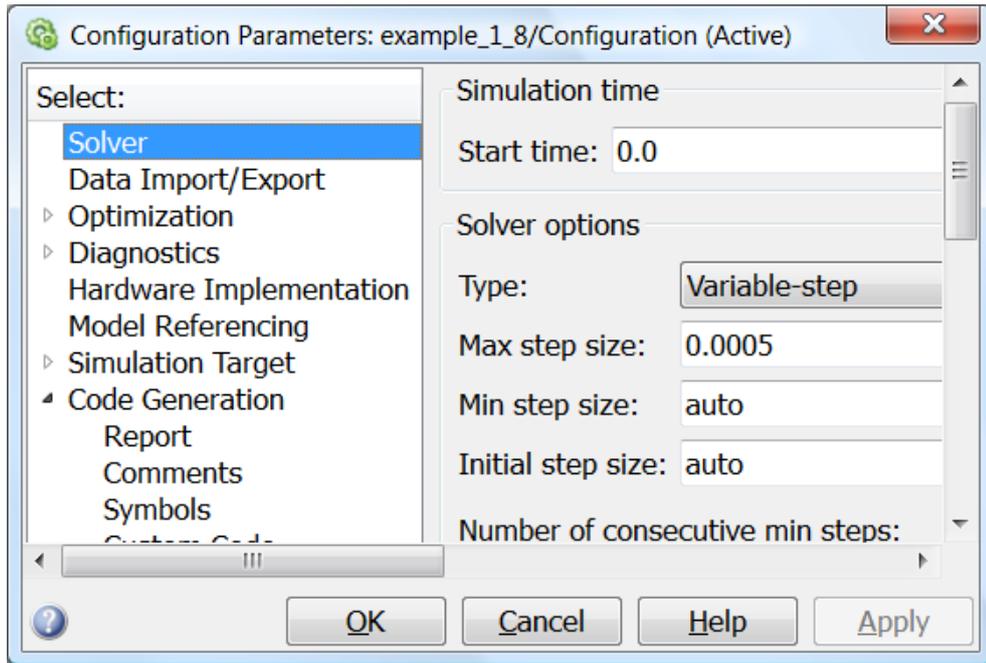


Рис. 1.26

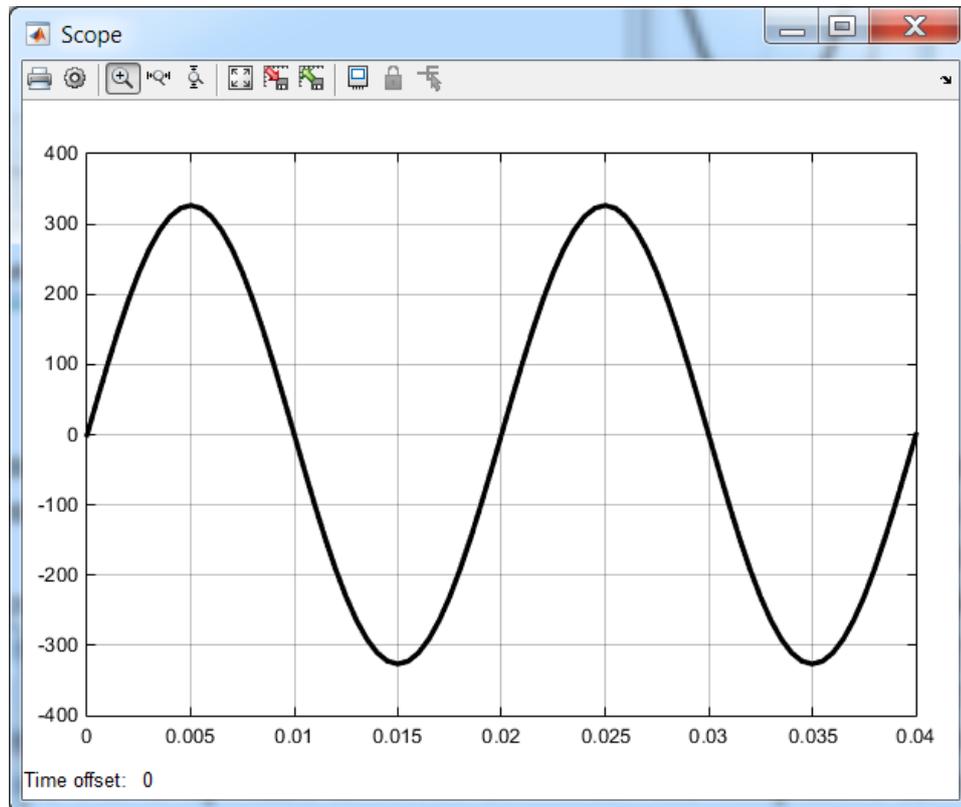


Рис. 1.27

В данном случае в окне осциллографа изображена черная кривая на светлом фоне, а не наоборот, как установлено по умолчанию. Для изменения стиля осциллографа необходимо нажать в панели инструментов кнопку  **Parameters**, затем перейти во вкладку **Style** и далее действовать по вкусу. Единственный минус – нет доступа к шрифтам, к названиям осей. Однако следует отметить, что в старых версиях **Simulink** такая "роскошь" в **Scope** вообще отсутствовала. В любом случае, как и прежде, имеется возможность сохранять данные в файлах. Подробности – в соседней вкладке **History** → **Save data to workspace**.

Для того, чтобы оценить достоинства расширения **Simulink** по сравнению с **MATLAB**, решим задачу *Примера 1.5* с помощью его блоков.

Пример 1.9.

Поскольку этот продукт изначально был предназначен для анализа систем автоматического управления, то представим дифференциальное уравнение трансформатора (1.1) в виде передаточной функции. В этой функции в качестве входного сигнала выступит напряжение, а в качестве выходного – ток короткого замыкания:

$$W_{Tp} = \frac{I_k(p)}{U_1(p)} = \frac{1}{r_k + L_k p} \quad (1.7)$$

или, как принято в **Simulink**,

$$W_{Tp} = \frac{1}{L_k s + r_k} = \frac{1}{(2.734e-5)s + 0.00192}. \quad (1.8)$$

При создании модели за основу возьмем предыдущий пример *example_1_8*, сохранив его под названием *example_1_9.slx* в новой папке *example_1_9*. Это позволит использовать в новой программе два блока из программы *example_1_8* с уже внесенными в них параметрами.

Для решения (1.8) воспользуемся блоком передаточной функции **Transfer Fcn** из раздела **Continuous**, в которую внесем данные из (1.8). Установим время окончания расчета равным 0,1 с.

В окне параметров **Source Block Parameters: Sine Wave** блока **Sine Wave** внесем фазу $\alpha = \psi_k + \pi/2 = \arctg(\omega L_k / r_k) = 1.5362 + \pi/2$, при которой наблюдается максимальное значение апериодической составляющей тока короткого замыкания.

Окончательный вид **Simulink** модели для исследования короткого замыкания в трехфазном трансформаторе представлен на рис. 1.28, а результаты расчета – на рис. 1.29. Эти результаты полностью совпали с теми, которые были получены с помощью программы *example_1_5* (рис. 1.16), разработанной в системе **MATLAB**.

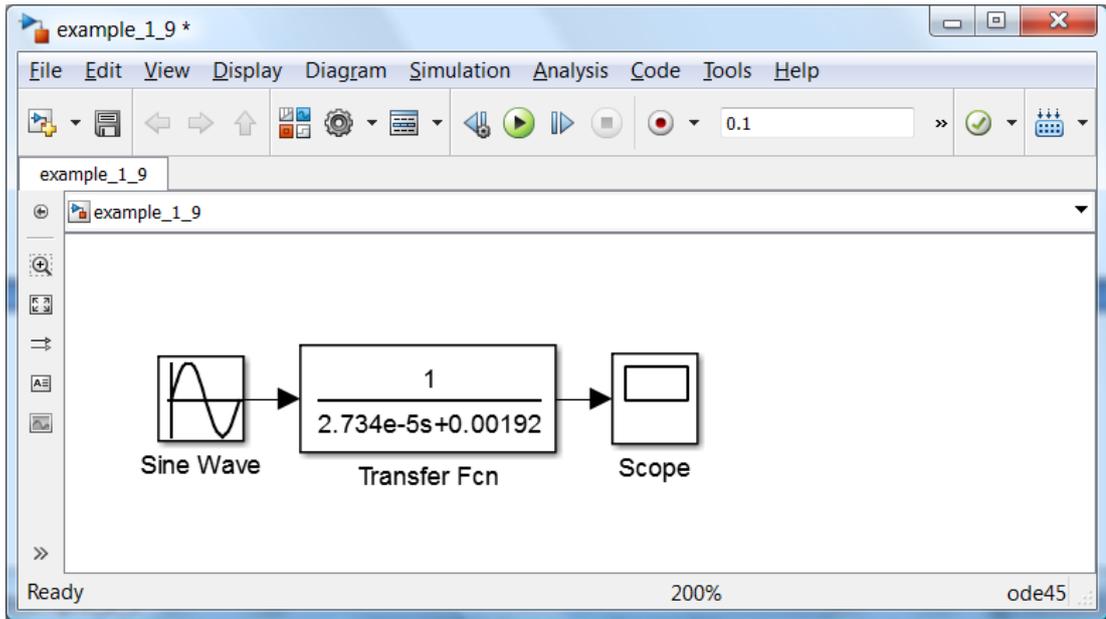


Рис. 1.28

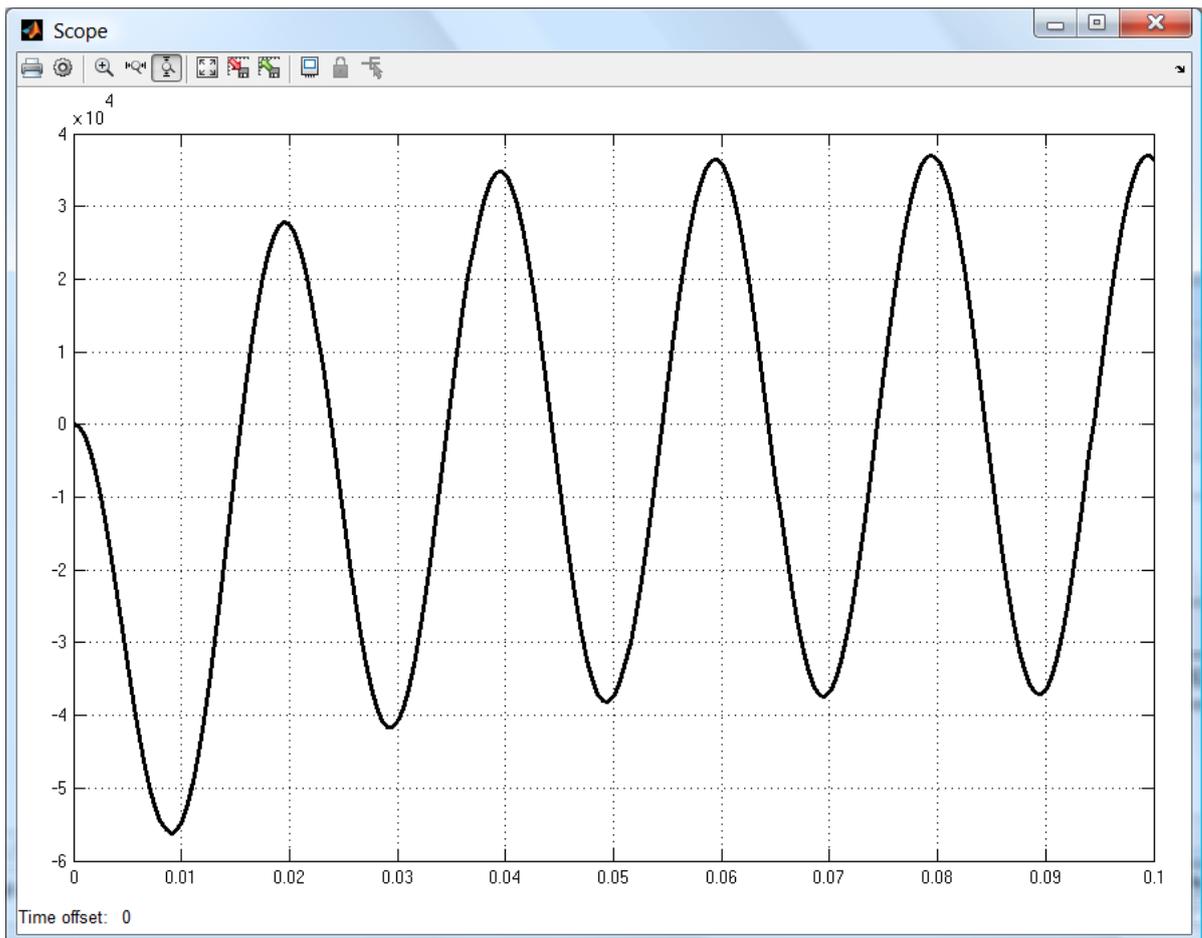


Рис. 1.29

Таким образом, для создания модели расчета тока короткого замыкания трехфазного трансформатора с помощью **Simulink** потребовались три блока, соединенных между собой, в два из которых следует внести исходные данные. Всё просто и наглядно. Нет никаких функций, строк кода и иных дополнительных действий, как при разработке m-файла в системе **MATLAB**.

Следует подчеркнуть, что **Simulink** предоставляет и другие способы решения системы дифференциальных уравнений, например, путем составления структурной схемы математической модели.

Воспользуемся таким подходом при моделировании ДПТ независимого (параллельного) возбуждения.

Пример 1.10.

Заменяем $d/dt \leftrightarrow s$ (в отечественной литературе – $d/dt \leftrightarrow p$) и перепишем (1.3) в следующем виде:

$$\left. \begin{aligned} i &= (u - c\omega) \left(\frac{1}{Ls + R} \right); \\ \omega &= (ci - M_c) \left(\frac{1}{Js + 0} \right). \end{aligned} \right\} \quad (1.9)$$

Модель для решения системы (1.9) набирается путем перетаскивания блоков из соответствующих библиотек в окно модели (рис. 1.30). Выбираем раздел **Continuous**, выделяем блок **Transfer Fcn**, нажимаем левую кнопку мыши и перетаскиваем блок в окно модели. Поскольку в модели используются два таких блока, то операцию можно повторить или скопировать этот блок в окне модели, используя способы, которые применяются при копировании файлов в ОС Windows. Из библиотеки **Math Operations** потребуются блоки **Gain** (2 шт.) и **Sum** (2 шт.), а из библиотек **Sinks** и **Sources** соответственно блоки **Scope** (2 шт.), **Step** (2 шт.), блок **Display**, а также из раздела **Signal Routing** блок **Mux**.

Для решения первого уравнения системы (1.9) значение напряжения u заносим в **Step**, который соединяем с сумматором **Sum**, в котором осуществляется операция $(u - c\omega)$. Блок **Sum** соединяем с блоком **Transfer Fcn**, осуществляющим операцию $1/(Ls + R)$. Для изменения знака сигнала в сумматоре **Sum** необходимо два раза щелкнуть левой кнопкой мышки на **Sum**. В открывшемся окне в строчке **List of signs** изменить последний «+» на «-» и закрыть окно.

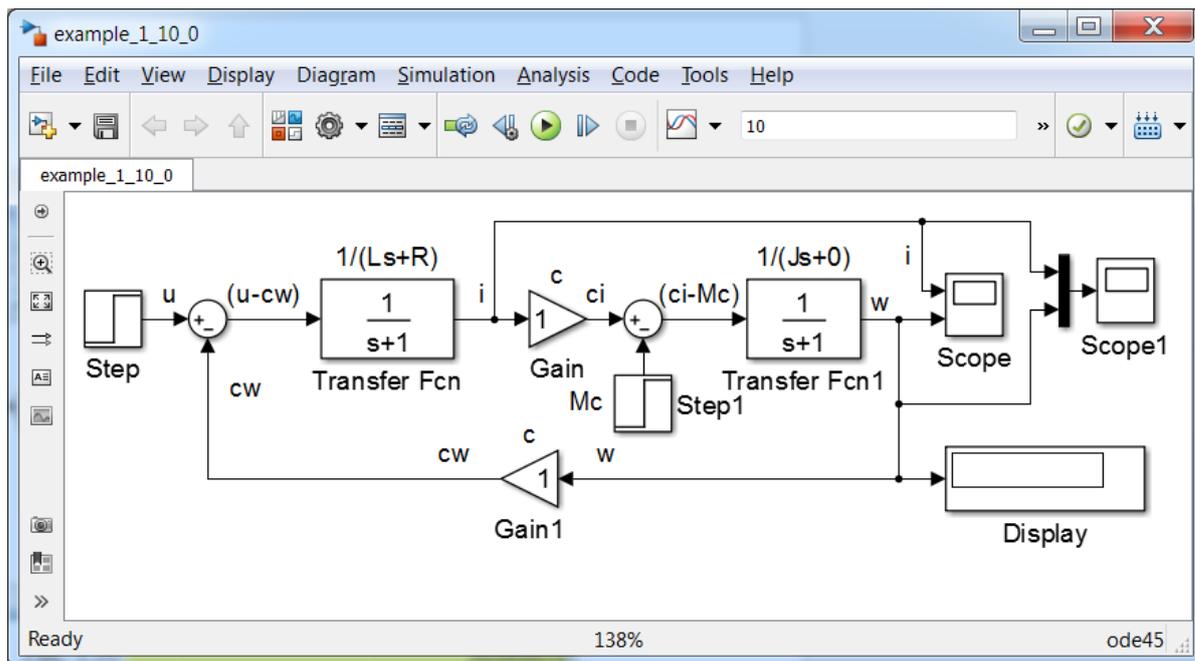


Рис. 1.30

Аналогичные изменения следует внести и в блок **Sum 1**.

Для решения второго уравнения системы (1.9) ток i на выходе блока **Transfer Fcn** умножаем на c (с помощью блока **Gain**) и подаем на сумматор **Sum 1**. Изменение позиции входа M_c осуществляем путем введения в поле **List of signs** следующих символов $- + |$. Так же для осуществления операции $(ci - M_c)$ на **Sum 1** подаем момент нагрузки M_c (возмущение), значение которого задаем в блоке **Step 1**. Блок **Sum 1** соединяем с блоком **Transfer Fcn 1**, который реализует операцию $1/(Js + 0)$. На выходе этого блока получаем угловую скорость ротора ω , значение которой подаем на осциллограф **Scope** и на блоке **Gain 1**. После умножения в этом блоке скорости ω на коэффициент c передаем полученный результат в блок **Sum**. Для изменения направления входа и выхода блока **Gain 1** можно нажать **Ctrl+I** или выделить блок, нажать правую кнопку мышки и перейти **Format** \rightarrow **Flip Block**. Для создания узла на линии связи необходимо подвести курсор мышки к этой линии, нажать правую кнопку и соединить новую линию с входом блока, например, с входом блока **Gain 1**.

Полученная структурная схема (рис. 1.30) полностью соответствует системе уравнений ДПТ (1.9). Однако она не позволяет исследовать характеристики объекта, пока его параметры не внесены в модель. Предположим, что технические данные и параметры ДПТ в номинальном режиме равны параметрам двигателя из *Примера 1.6*. В этом случае вносим эти значения в соответствующие окна параметров (Рис. 1.32).

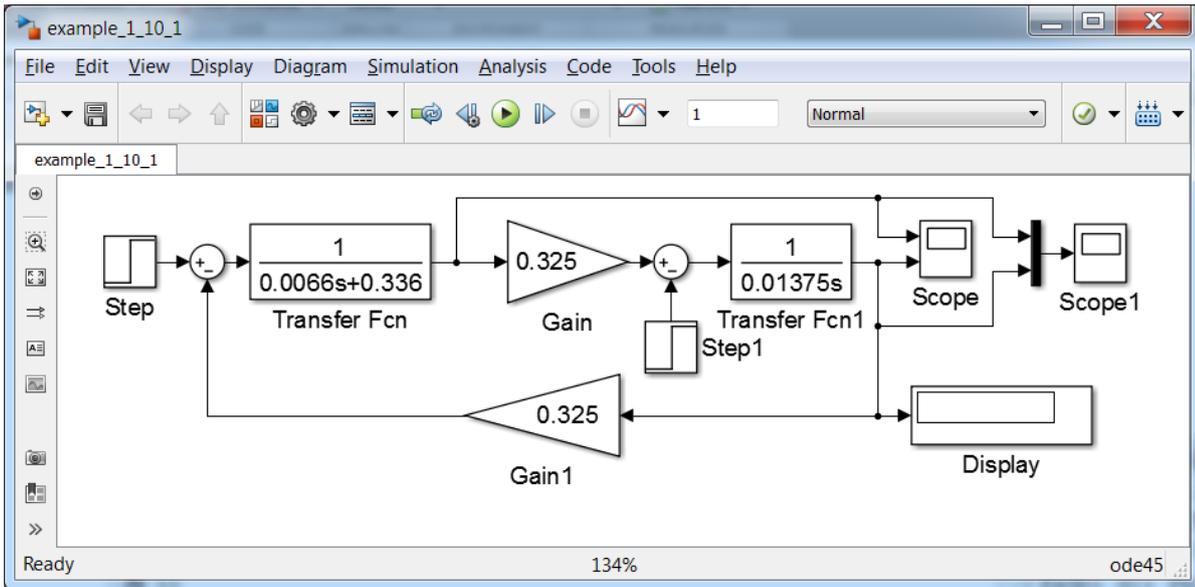


Рис. 1.31

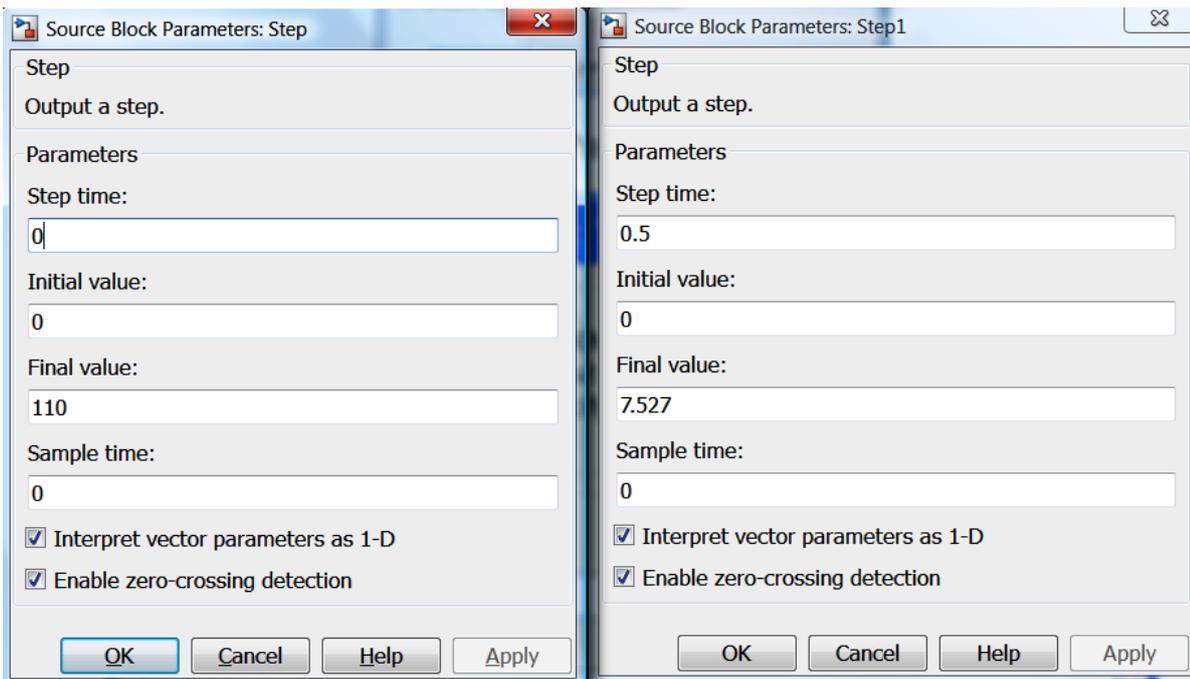


Рис. 1.32

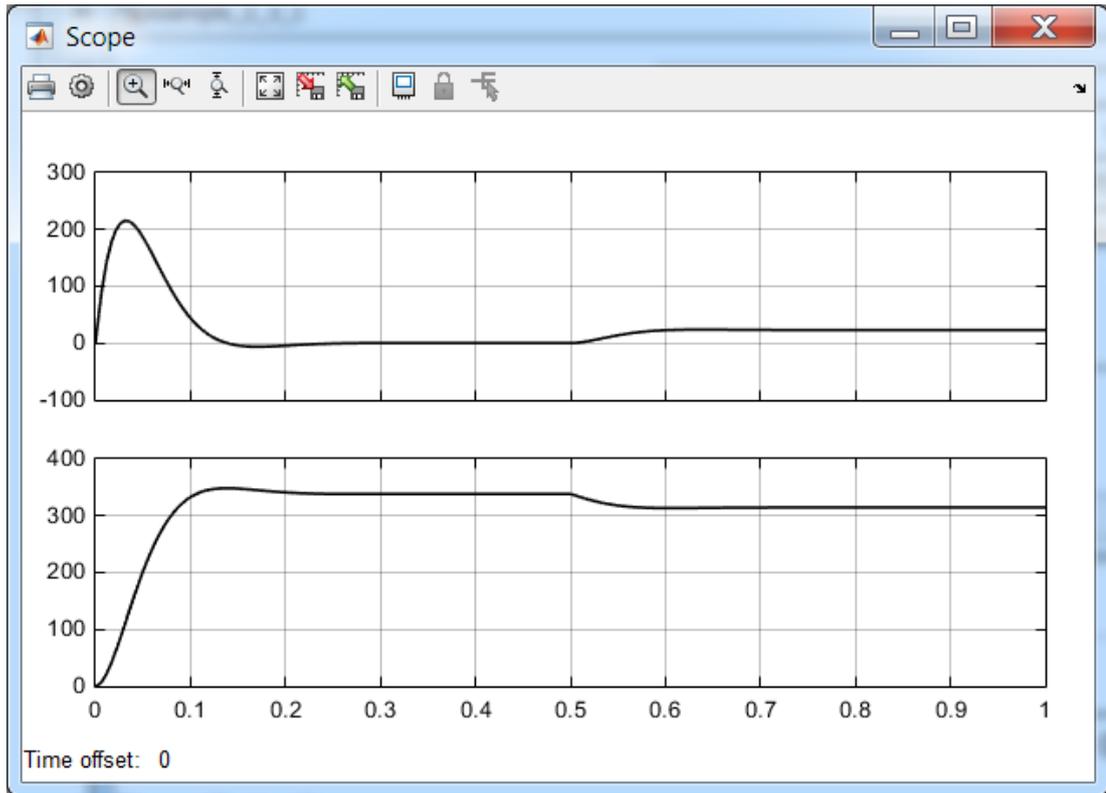


Рис. 1.32

Если эти численные значения не отображаются в блоках (рис. 1.31), то их необходимо увеличить, выделив блок и потянув его за угол, как это обычно делают с рисунками или фотографиями в редакторе.

В поле ввода **Simulation stop time** панели инструментов (рядом с кнопкой  **Run**) вводим время расчета $t=1$ (рис. 1.31). Можно воспользоваться и командой меню **Simulation** → **Configuration Parameters** → **Stop time**. В поле **Step time** блока **Step 1**, вводим время наброса нагрузки M_C (возмущения), равное 0,5 с.

После этого запускаем модель, нажав кнопку  **Run**. Двойным щелчком левой кнопки мышки на **Scope** (осциллограф) визуализируем результаты расчета (рис. 2.11). Как и следовало ожидать, эти результаты совпали с теми, которые представлены на рис. 1.17. Однако, на рис. 1.32 каждая кривая выведена в отдельной координатной плоскости, в первом – момент, а во втором – ток.

В принципе, количество сигналов, которые могут быть выведены с помощью **Scope**, превосходит разумную необходимость. Для изменения количества сигналов необходимо открыть **Scope**, на панели инструментов **Scope** нажать кнопку  **Parameters**, затем перейти во вкладку **General** и ввести необходимое количество осей в окошко **Number of axes** (рис. 1.33).

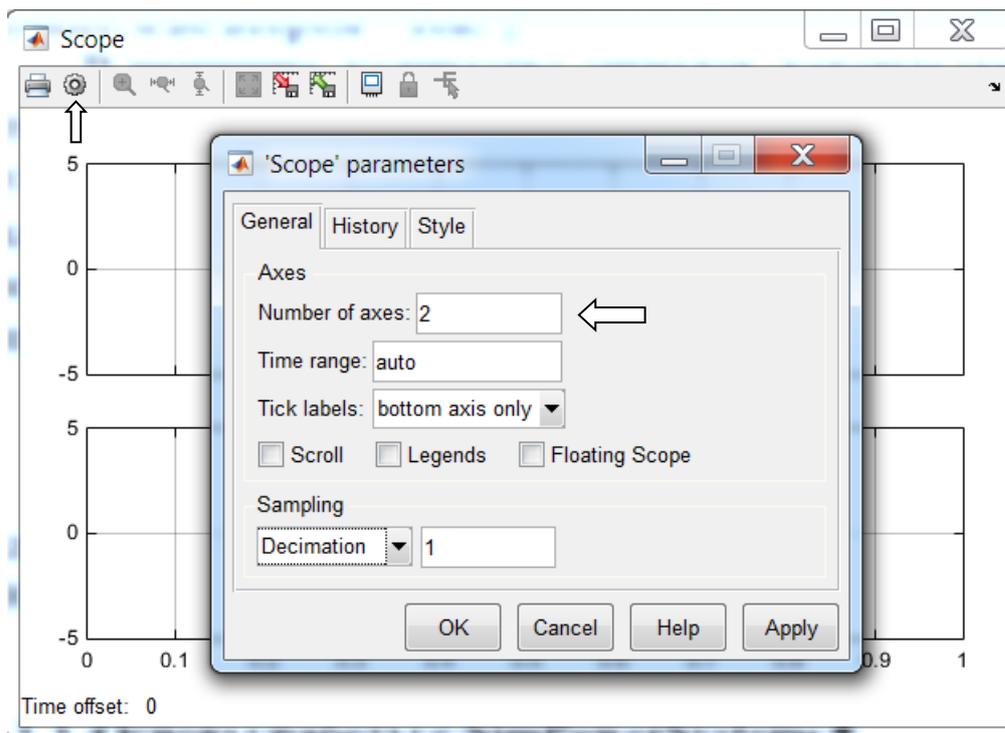


Рис. 1.33

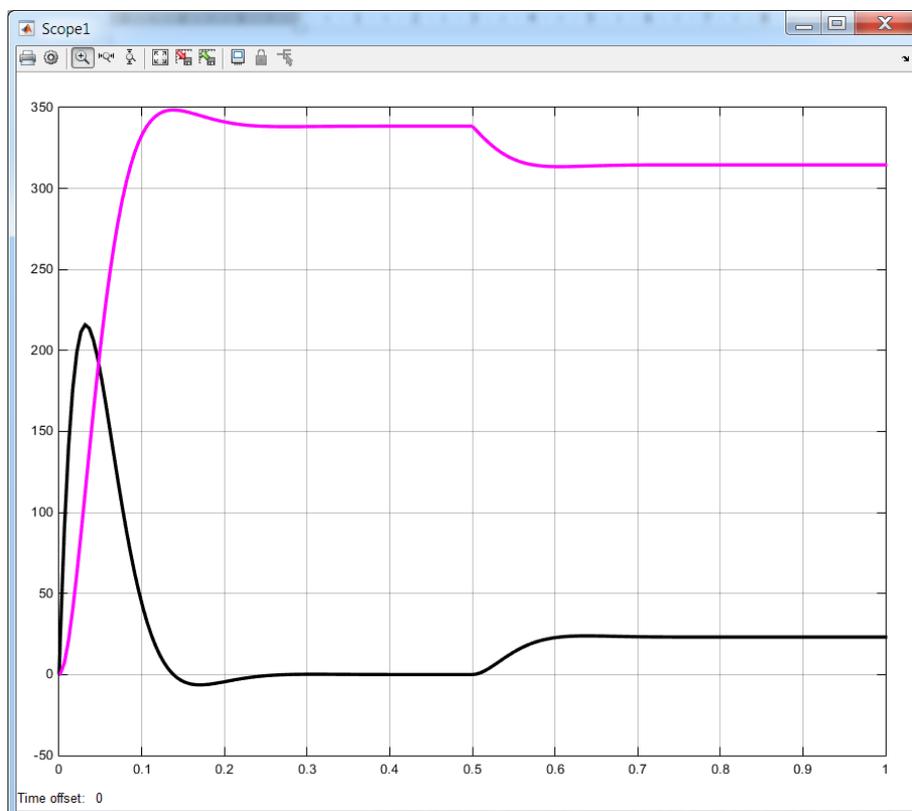


Рис. 1.34

Для того, чтобы вывести несколько сигналов в одной координатной плоскости, необходимо их смешать с помощью блока **Mux** из раздела **Signal Routing** и подать этот сигнал на вход осциллографа, в нашем случае – **Scope 1**. При этом получим графики, представленные на рис. 1.34. В блоке **Display** (Рис. 1.31) показано значение скорости ротора в установившемся режиме (в конце расчета).

Очень часто, при подготовке отчетов, возникает необходимость переноса полученных графиков в текстовый редактор, например, в **Word**. В этом случае, если к качеству графиков не предъявляется особых требований, следует выделить график **Scope**, нажать сочетание клавиш **Alt+PrtScn**, перейти в редактор и вставить изображение в документ, нажав **Ctrl+V**.

При более жестких требованиях к качеству графиков информацию следует записать в переменную, затем с неё передать информацию в **Word** или **Excel**, а после строить графики, используя возможности этих программ. Для записи в переменную нажимаем кнопку  **Parameters**, переходим во вкладку **History**. Снимаем ограничение **Limit data points to last**, если вместо вывода заключительной части графика предполагается вывод всего графика, т.е. со значения $t = 0$ (**Start time**) до **Stop time**.

Активизируем **Save data to workspace**, вводим имя переменной, например, **ScopeData_example_1_10**, выбираем формат **Array** (рис. 1.35).

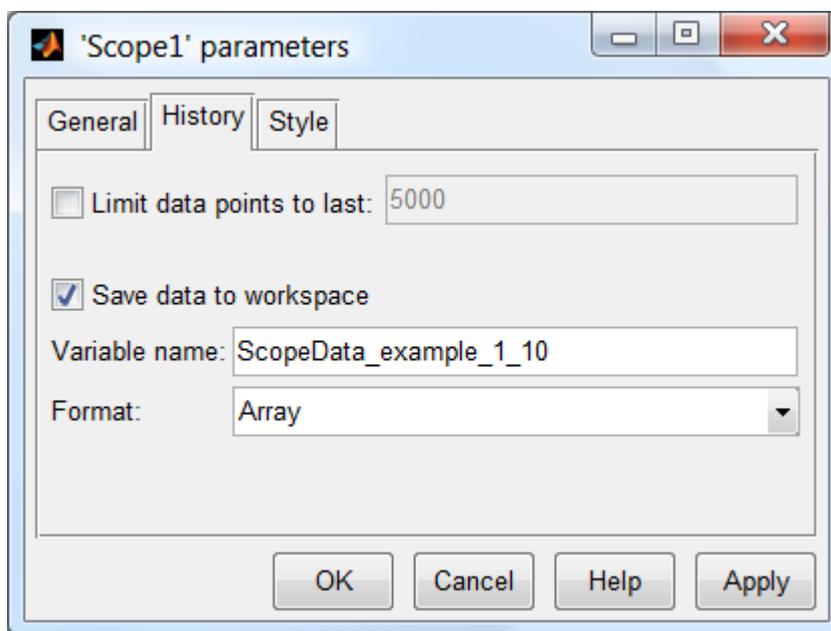


Рис. 1.35

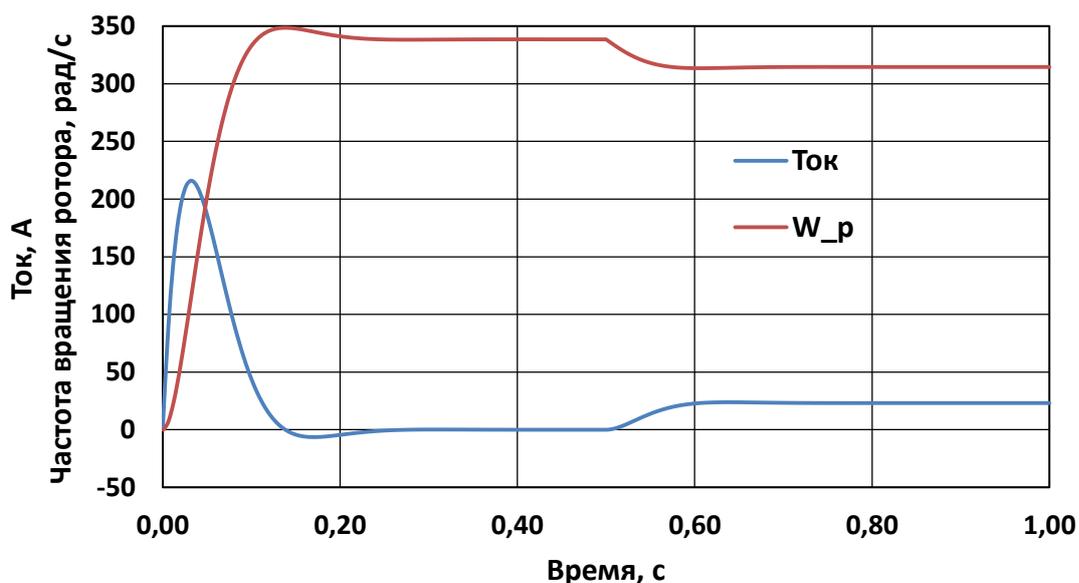


Рис. 1.36

После этого переходим в **MATLAB**. В окне **Workspace** нажимаем на переменную **ScopeData_example_1_10**. Копируем данные в «карман», нажав **Ctrl+C**. Запускаем **Word**. Во вкладке **ВСТАВКА** нажимаем **Диаграмма**, выбираем **Точечная** → **Точечная с гладкими кривыми**. Вставляем в её таблицу данные из **MATLAB**, нажав **Ctrl+V**. Далее выполняем стандартные действия по оформлению диаграммы, например, вводим название осей, легенду, подгоняем оси, изменяем шрифты и т.д. В результате получаем диаграмму, гармонично вписывающуюся в стиль редактора **Word** (Рис. 1.36).

1.3. SimPowerSystems

1.3.1. Система моделирования SimPowerSystems

Для моделирования электроэнергетических систем, а также устройств электротехники и промышленной электроники служит пакет расширения **SimPowerSystems**. В этом пакете имеются блоки большинства электротехнических элементов и устройств: трансформаторов, ЛЭП, синхронных генераторов, асинхронных двигателей, полупроводниковых элементов, преобразователей, измерительных приборов и т.д.

Методика построения моделей **SimPowerSystems** практически не отличается от методики построения **Simulink** моделей. И в том, и в другом случае используются блоки, которые соединяют между собой. Отличие же заключается в том, что в моделях **Simulink** это линии передачи сигнала, а в моделях **SimPowerSystems** линии – некие

виртуальные проводники, по которым протекает виртуальный ток. Для измерения этого тока используется блок измерителя тока, который подключается в разрыв проводника (линии), а сигнал с этого блока поступает на осциллограф, дисплей или иной блок расширения **Simulink**.

Таким образом, эти два блока очень тесно связаны между собой. Особое внимание необходимо уделить изучению обширной библиотеки **SimPowerSystems**.

1.3.2. Создание, редактирование и запуск модели SimPowerSystems

Рассмотрим, как можно решить задачу *Примера 1.5* с помощью блоков **SimPowerSystems**.

Пример 1.11.

Запускаем **MATLAB**, нажимаем на панели инструментов кнопку **Library Browser** , в панели инструментов библиотеки нажимаем на кнопку  **New model**. При этом создается пустое окно с названием файла **untitled**. Сохраним это окно в новой папке, например, *example_1_11*, под названием *example_1_11*, с расширением файла *.slx или *.mdl. Для этого нажимаем на панели инструментов **File** → **Save As...** и т.д. Следует еще раз напомнить, что использование кириллицы в названии файла, папки или во всём пути к файлу может стать причиной появления сообщения об ошибке при сохранении модели в формате *.mdl (*Не совпадают кодовые страницы операционной системы и MATLAB*).

После этого переходим к окну библиотеки блоков **Library Browser**, если оно не перекрыто другими окнами. В противном случае, на панели инструментов окна модели следует нажать кнопку  **Library Browser** и закрепить его поверх остальных открытых окон, нажав на кнопку  **Stay on top**.

В списке разделов библиотеки переходим к папке **Simscape** → **SimPowerSystems** → **Specialized Technology** → **Electrical Sources** (Рис. 1.37).

В этом разделе (папке) выбираем источник переменного напряжения **AC Voltage Source** (Alternate Current – переменный ток) и перетаскиваем его в окно модели. Для измерения тока нам потребуется блок **Current Measurement**, который находится в разделе **Measurements** (Рис. 1.38). Этот блок также перетаскиваем в окно модели.

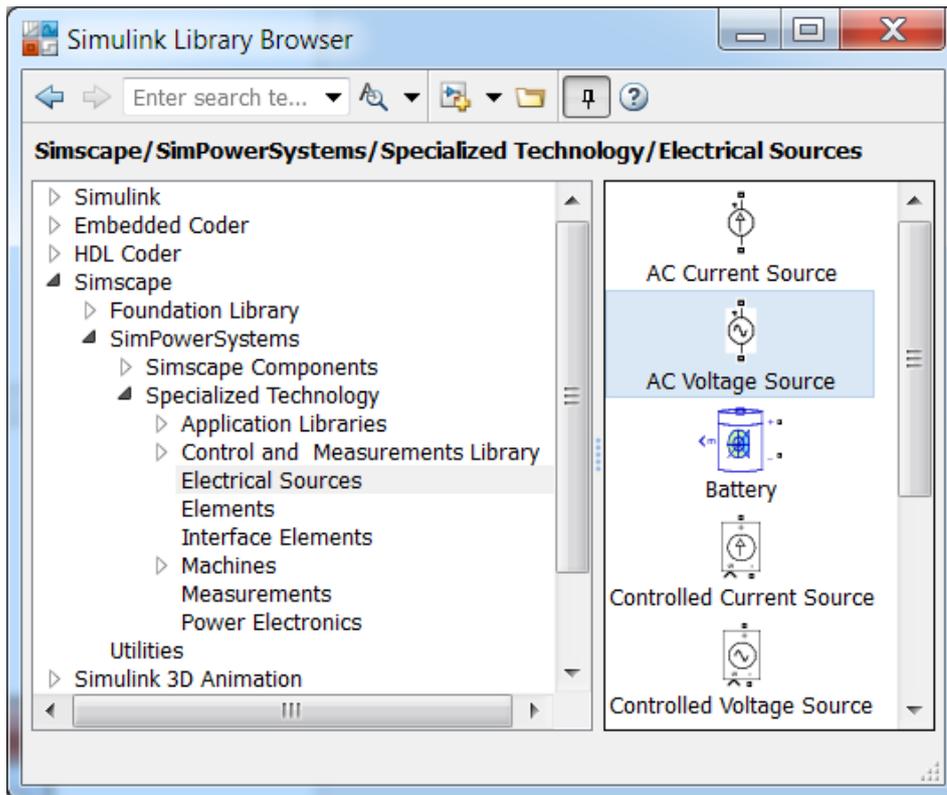


Рис 1.37

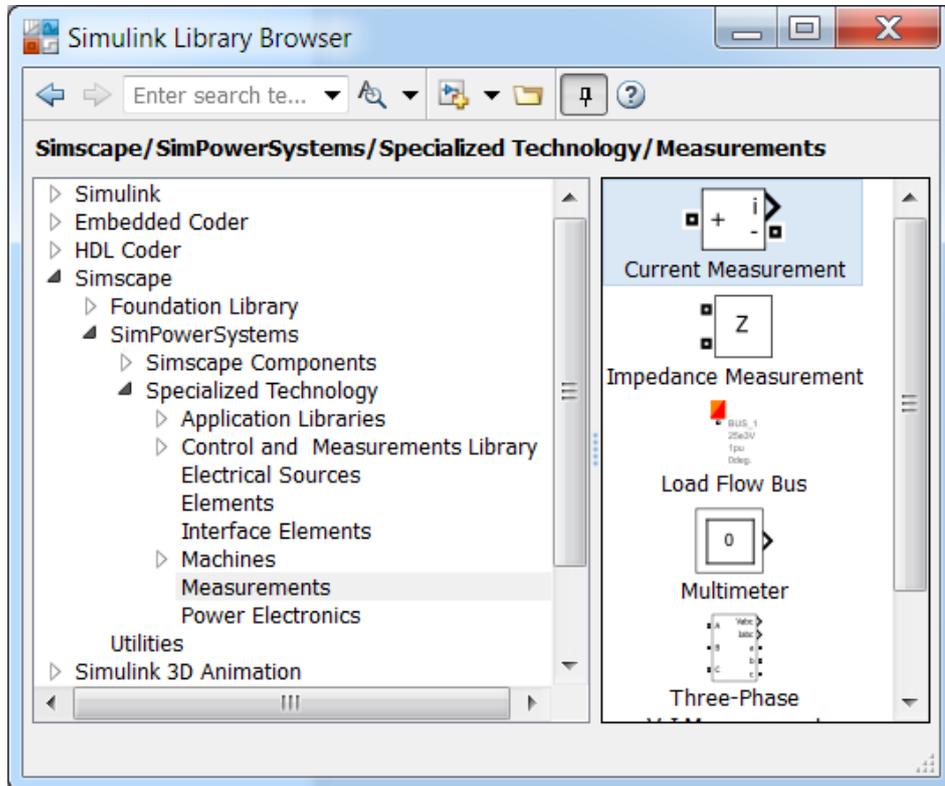


Рис 1.38

Блок измерения тока **Current Measurement** при его включении в цепь **SimPowerSystems** вырабатывает сигнал, который может подаваться на любой приемник расширения **Simulink**. Остановим свой выбор на блоке **Scope** и перенесем его из библиотеки в окно модели.

В данном случае ток короткого замыкания ограничивается только активно-индуктивным сопротивлением короткого замыкания трансформатора. В связи с этим, при создании модели воспользуемся блоком **Series RLC Branch** (последовательная RLC ветвь), который находится в разделе **Elements** (Рис. 1.39).

Поскольку блоки **SimPowerSystems** отличаются от блоков **Simulink**, то отличаются и способы их подключения. Так, в блоках **Simulink** выход и вход обозначены треугольником "▷" и знаком больше ">", т.е. сигнал передается от выхода к входу. В блоках **SimPowerSystems** условные клеммы для подключения блоков **SimPowerSystems** обозначены квадратом "□".

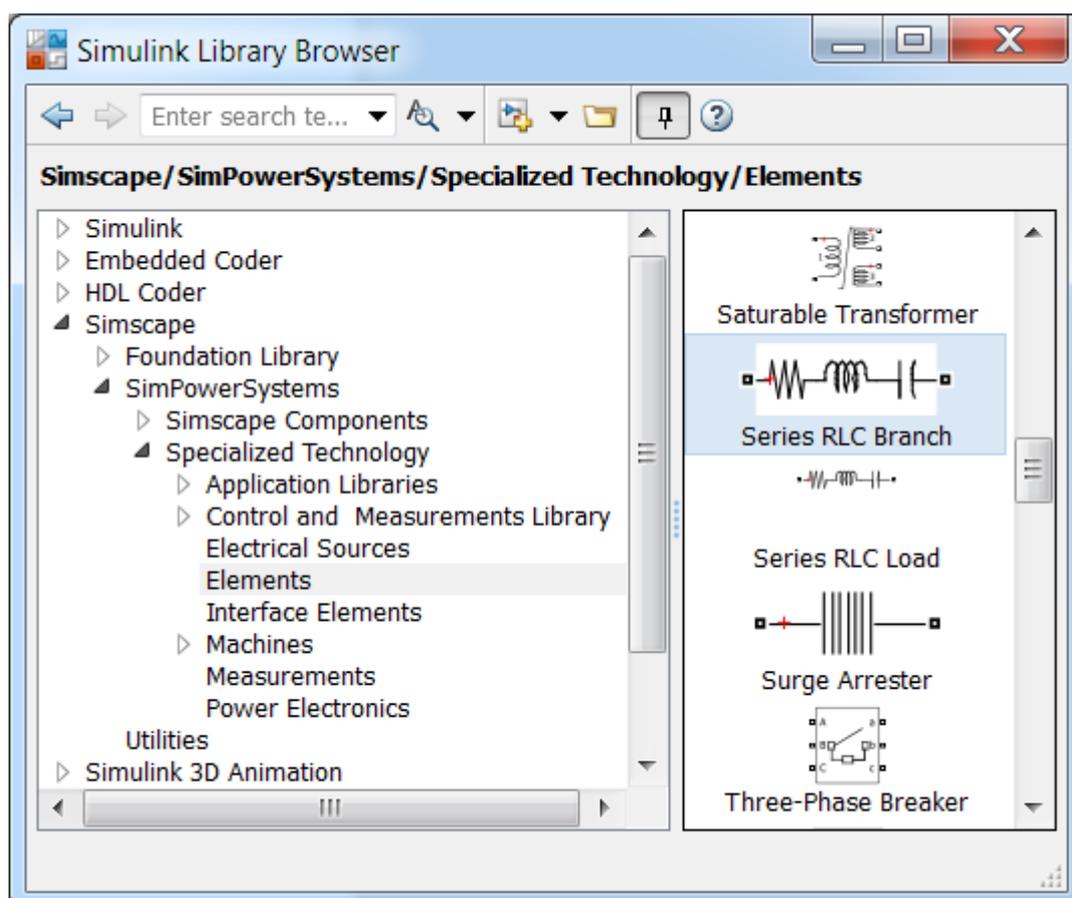


Рис 1.39

После того, как все блоки соединены, вводим из *Примера 1.5* в соответствующие блоки значения напряжения сети и параметры короткого замыкания трансформатора (Рис. 1.40). Устанавливаем время расчета 0,1 с. Для работы модели необходимо установить ещё один блок, который называется **Powergui block** и находится в разделе **Specialized Technology**. Как и в случае с **Simulink** необходимо установить в решателе максимальный шаг равный 0,0005 с (Рис. 1.26).

В итоге получается модель, показанная на рис. 1.41. Следует отметить, что результаты работы этой модели (Рис. 1.42) совпали с результатами расчета, которые были осуществлены средствами **MATLAB** и **Simulink**, представленными соответственно на рис. 1.16 и рис. 1.29.

Сравнительный анализ всех трех подходов показывает, что наиболее комфортным является моделирование в расширениях **Simulink** и **SimPowerSystems**, т.к. в них процесс создания модели сводится к элементарному соединению необходимых блоков, заданию их параметров и настройке решателя. В отличие от **Simulink**, блоки **SimPowerSystems** наиболее близки реальным элементам и объектам электротехники, что позволяет говорить о его неоспоримых преимуществах при моделировании электротехнических систем и устройств.

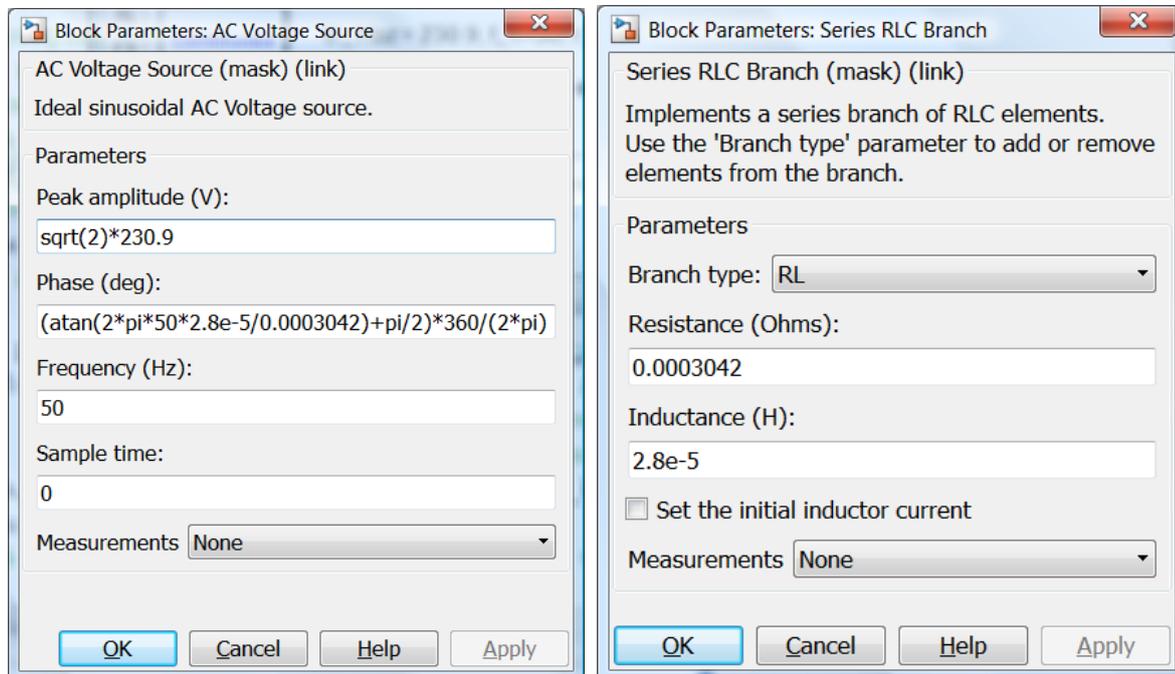


Рис. 1.40

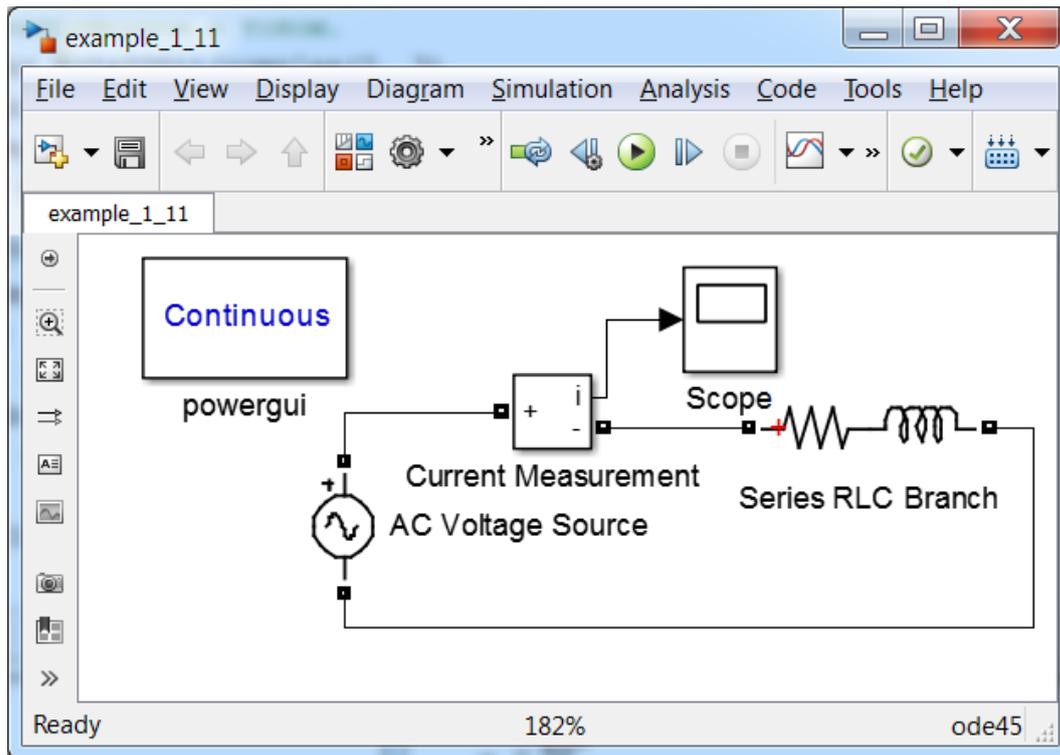


Рис. 1.41

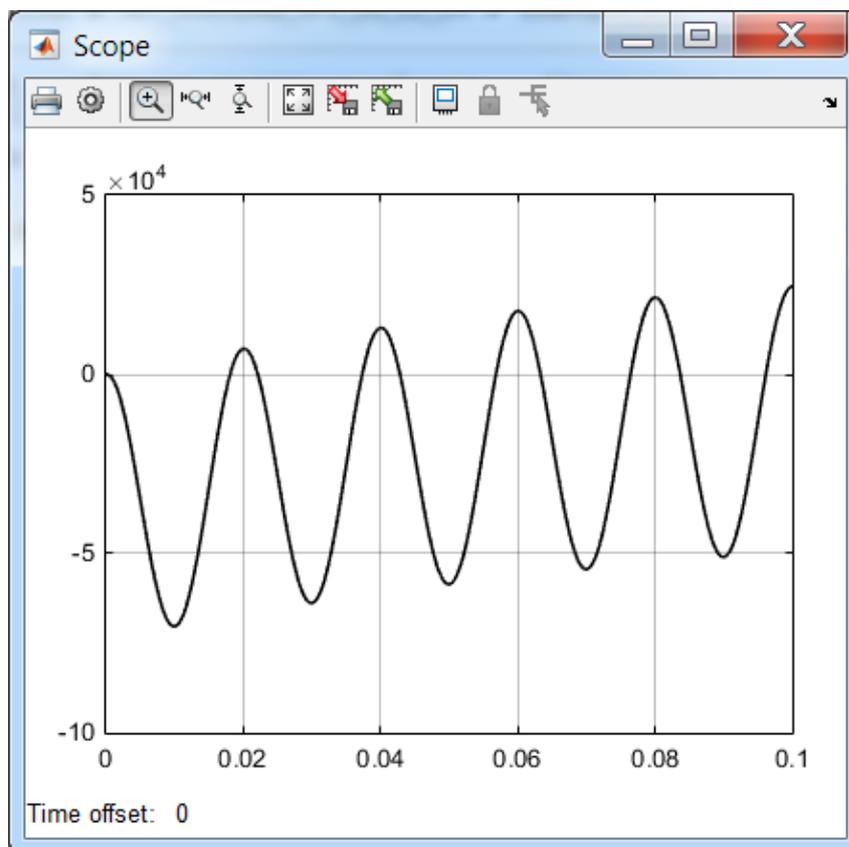


Рис. 1.42

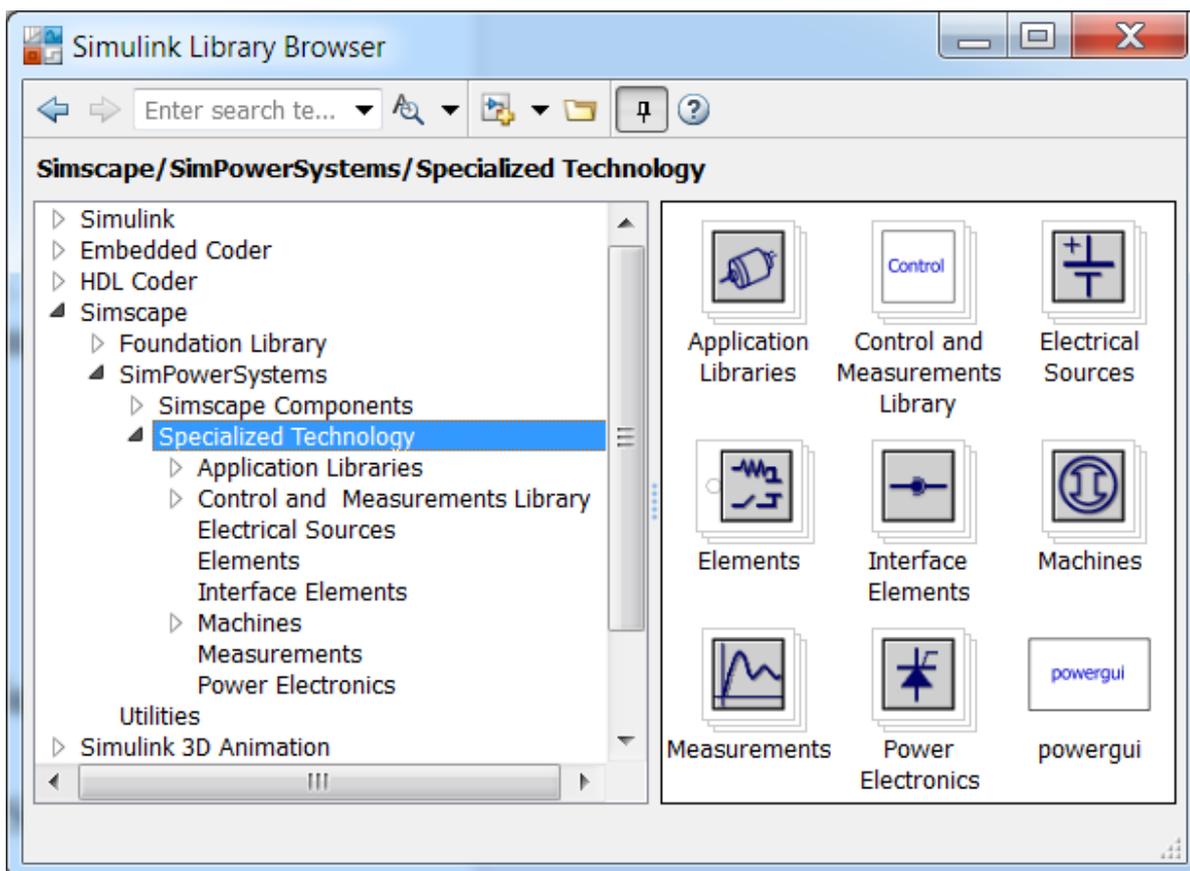


Рис. 1.43

При оценке эффективности любого стандартного языка программирования определяющее значение имеет количество доступных функций в его библиотеках. В этом плане эффективность **MATLAB** не вызывает сомнений, поскольку расширения системы **MATLAB+Simulink**, которых более 80, имеют в своем составе огромное количество специализированных блоков, позволяющих создать модель всего объекта с учетом взаимодействия его узлов различной физической природы.

Например, в модели автомобиля с помощью расширений **SimMechanics**, **SimDriveline** и **SimPowerSystems** учитывается взаимодействие механических и электрических узлов. Далее кратко рассмотрим библиотеку блоков расширения **SimPowerSystems Specialized Technology** (Рис. 1.43).

1.3.3. Состав библиотек SimPowerSystems Specialized Technology

В состав библиотеки входят следующие разделы (рис. 1.43):

1) **Application Libraries** – библиотеки блоков прикладного характера представлены на рис. 1.44 (Глубина раскрытия подпунктов библиотек зависит от актуальности представленных в них блоков):

- ✓ **Electric Drives library** – библиотека электроприводов [7 , 8 , 9]:
 - **AC drives** – электроприводы переменного тока:
 - **Brushless DC Motor Drive** – электропривод на базе бесконтактного (бесщеточного) двигателя постоянного тока, выполненного на основе синхронного двигателя с постоянными магнитами;
 - **DTC (direct torque and flux control) Induction Motor Drive** – электропривод, в котором обеспечивается прямое управление моментом и полем асинхронного двигателя;

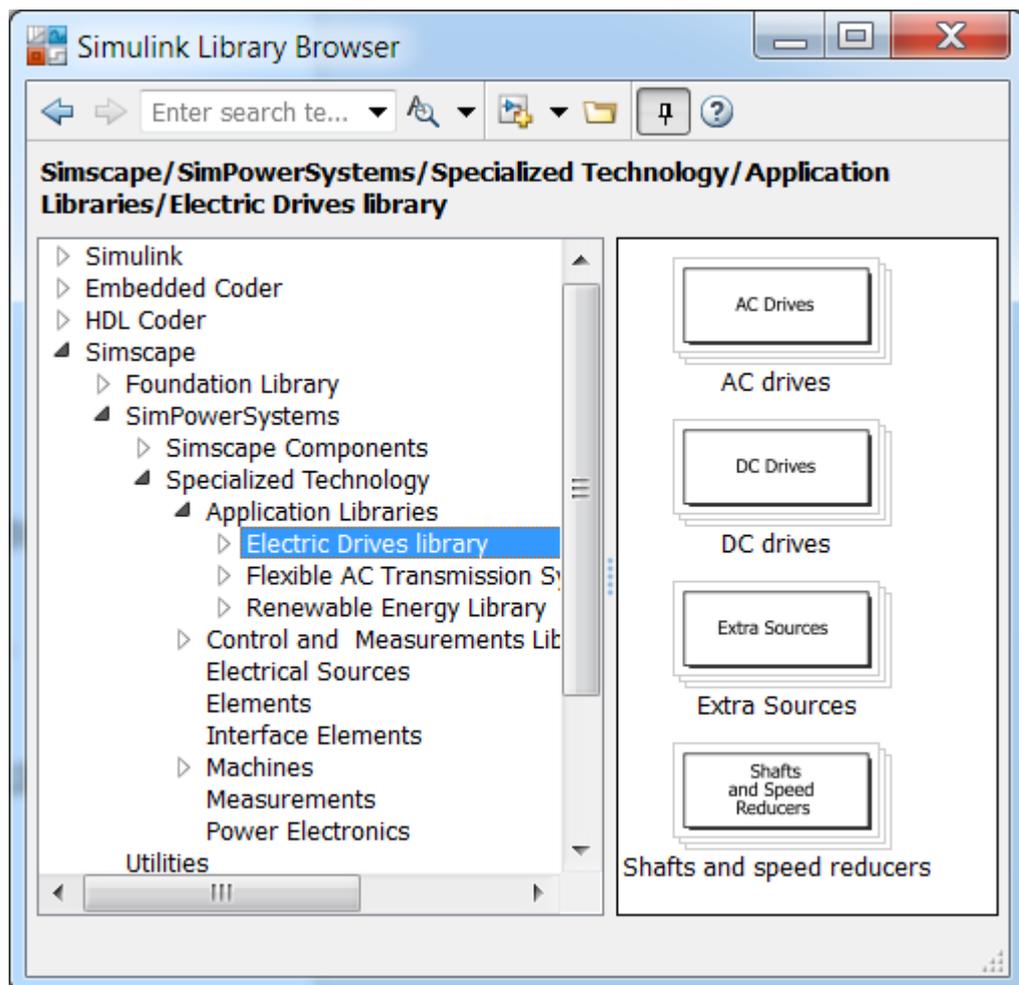


Рис. 1.44

- **Field-Oriented Control Induction Motor Drive** – электропривод, с поле-ориентированным управлением асинхронного двигателя;
- **Five-Phase PM Synchronous Motor Drive** – электропривод с векторным управлением 5-ти фазного синхронного двигателя с постоянными магнитами;
- **PM Synchronous Motor Drive** – электропривод с векторным управлением синхронного двигателя с постоянными магнитами;
- **Self-Controlled Synchronous Motor Drive** – электропривод с вентильным синхронным двигателем;
- **Six-Step VSI (voltage source inverter) Induction Motor Drive** – электропривод с асинхронным двигателем на основе шестишагового инвертора напряжения;
- **Space Vector PWM (pulse width modulation) VSI (voltage source inverter) Induction Motor Drive** – электропривод с векторным управлением асинхронного двигателя на основе инвертора напряжения с широтно-импульсной модуляцией.
- **DC drives** – электроприводы постоянного тока [7]:
 - **Four-Quadrant Chopper DC Drive** – реверсивный электропривод постоянного тока с широтно-импульсным преобразователем (ШИП);
 - **Four-Quadrant Single-Phase Rectifier DC Drive** – реверсивный электропривод постоянного тока с однофазным выпрямителем;
 - **Four-Quadrant Three-Phase Rectifier DC Drive** – четырехквadrантный (реверсивный) электропривод постоянного тока с трехфазным выпрямителем;
 - **One-Quadrant Chopper DC Drive** – одно-квadrантный электропривод постоянного тока с ШИП;
 - **Two-Quadrant Chopper DC Drive** – двухквadrантный (нереверсивный) электропривод постоянного тока;
 - **Two-Quadrant Single-Phase Rectifier DC Drive** – двухквadrантный (нереверсивный) электропривод постоянного тока с однофазным выпрямителем;
 - **Two-Quadrant Three-Phase Rectifier DC Drive** – двухквadrантный (нереверсивный) электропривод постоянного тока с трехфазным выпрямителем.
- **Extra Sources** – дополнительные источники:
 - **Battery** – батарея; аккумулятор;
 - **Fuel Cell Stack** – батарея топливных элементов;
 - **Supercapacitor** – суперконденсатор.

- **Shafts and speed reducers** – валы и редукторы.
- ✓ **Flexible AC Transmission Systems (FACTS) Library** – гибкие системы передачи переменного тока:
 - **HVDC (High-Voltage Direct Current) Systems** – высоковольтные системы передачи постоянного тока. Для знакомства с моделями **HVDC** нажимаем клавишу **F1**, переходим в **SimPowerSystems**, выбираем **Examples** и нужную модель. При отсутствии **SimPowerSystems** после **F1** нажимаем на .
 - **Power-Electronics Based FACTS** – гибкие системы передачи переменного тока на базе силовой электроники:
 - **Static Synchronous Compensator (Phasor Type) (STATCOM)** – статический синхронный компенсатор (Векторный тип);
 - **Static Synchronous Series Compensator (Phasor Type) (SSSC)** – последовательный статический синхронный компенсатор (Векторный тип);
 - **Static Var Compensator (Phasor Type):** – статический компенсатор реактивной мощности (Векторный тип);
 - **Unified Power Flow Controller (Phasor Type)** – объединенный регулятор потока мощности (Векторный тип).
 - **Transformers** – трансформаторы с системой регулирования под нагрузкой (On-load tap changer – РПН).
- ✓ **Renewable Energy Library** – библиотека возобновляемых источников энергии:
 - **Wind Generation** – ветроэнергетическая установка (ВЭУ).

2) **Control and Measurements Library** – библиотека блоков контроля и измерений (Рис. 1.45):

- **Additional Components** – дополнительные компоненты;
- **Filters** – фильтры;
- **Logic** – блоки логики;
- **Measurements** – блоки измерений (Рис. 1.46):
 - **Fourier** – блок преобразования Фурье;
 - **Fundamental (PLL-Driven)** – блок определения основной гармоники путем использования фазовой автоподстройки частоты;
 - **Mean** – вычисляет среднее значение сигнала;
 - **Mean (Phasor)** – представляет среднее значение входного сигнала за один цикл основной гармоники в виде комплексного числа (*Phasor* – комплексное число, вектор на комплексной плоскости);

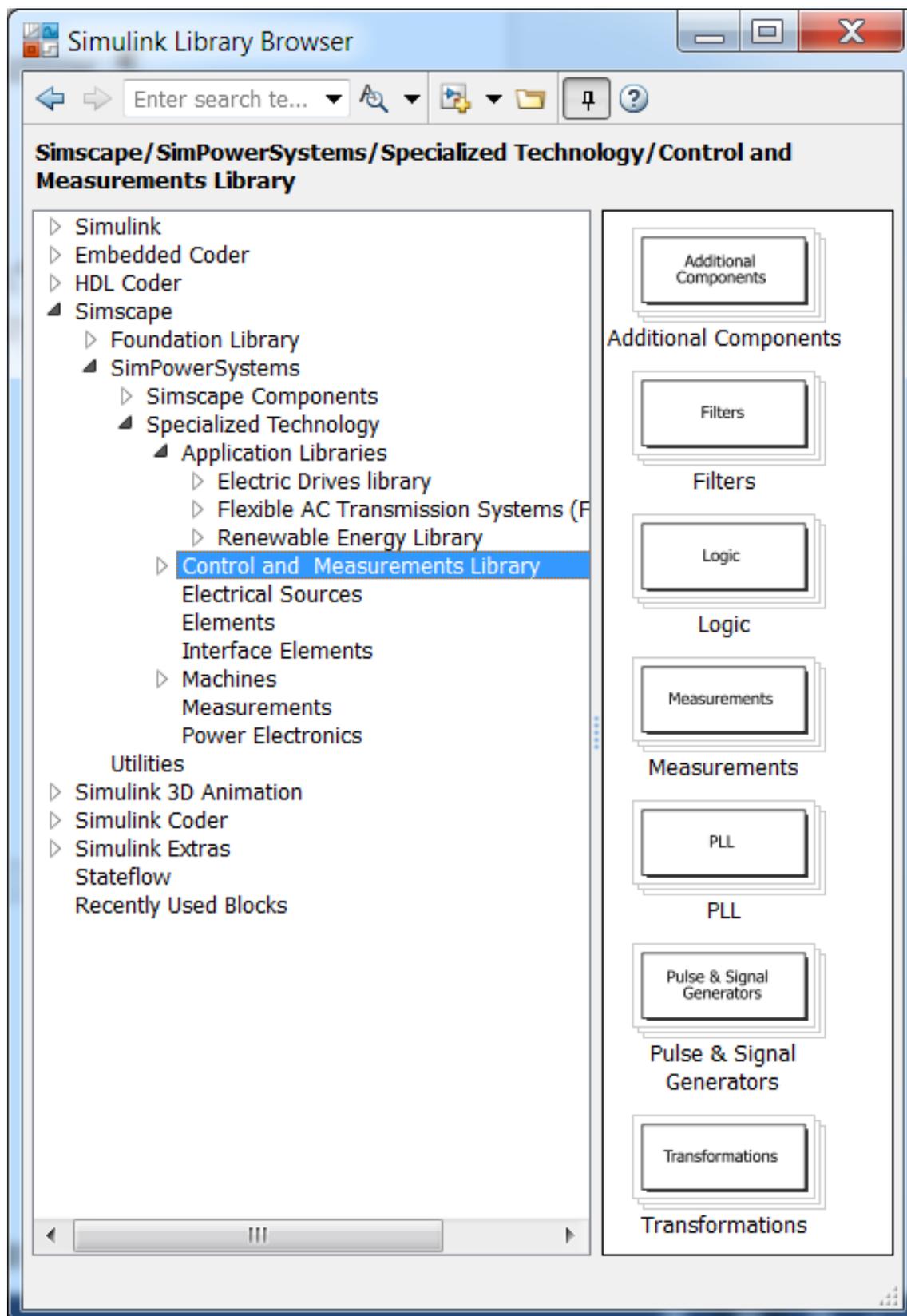


Рис. 1.45

- **Mean (Variable Frequency)** – вычисляет среднее значение сигнала за один период другого сигнала, подаваемого на второй вход блока;
- **Positive-Sequence (PLL-Driven) (Phase Locked Loop)** – вычисляет положительную (прямую) последовательность путем использования системы фазовой автоподстройки частоты;
- **Power** – блок вычисления активной и реактивной мощностей по значениям синусоидальных тока и напряжения за один период;
- **Power (3ph, Instantaneous)** – блок вычисления мгновенных (*Instantaneous*) значений активной и реактивной мощностей трехфазной симметричной синусоидальной системы токов и напряжений;
- **Power (3ph, Phasor)** – блок вычисления активной и реактивной мощностей по трехфазным комплексным значениям тока и напряжения;
- **Power (PLL-Driven, Positive-Sequence)** – блок рассчитывает активную и реактивную мощности положительной (прямой) последовательности токов и напряжений по частоте, определяемой с помощью фазовой автоподстройки;
- **Power (Phasor)** – блок вычисляет активную и реактивную мощности тока и напряжения в виде комплексного числа (сигнала);
- **Power (Positive-Sequence)** – блок рассчитывает активную и реактивную мощности положительной (прямой) последовательности токов и напряжений за период, установленный для этих синусоидальных трехфазных токов и напряжений;
- **Power (dq0, Instantaneous)** – рассчитывает мгновенные значения мощностей в осях dq с учетом нулевой последовательности;
- **RMS (root mean square)** – среднеквадратичное (действующее) значение сигнала (например, напряжения или тока);
- **Sequence Analyzer** – блок вычисляет прямую, обратную и нулевую последовательности трехфазного сигнала;
- **Sequence Analyzer (Phasor)** – блок вычисляет прямую, обратную и нулевую последовательности при подаче на вход трех сигналов в виде комплексных чисел;
- **THD (Total Harmonic Distortion)** – блок, рассчитывающий коэффициент гармонических искажений.

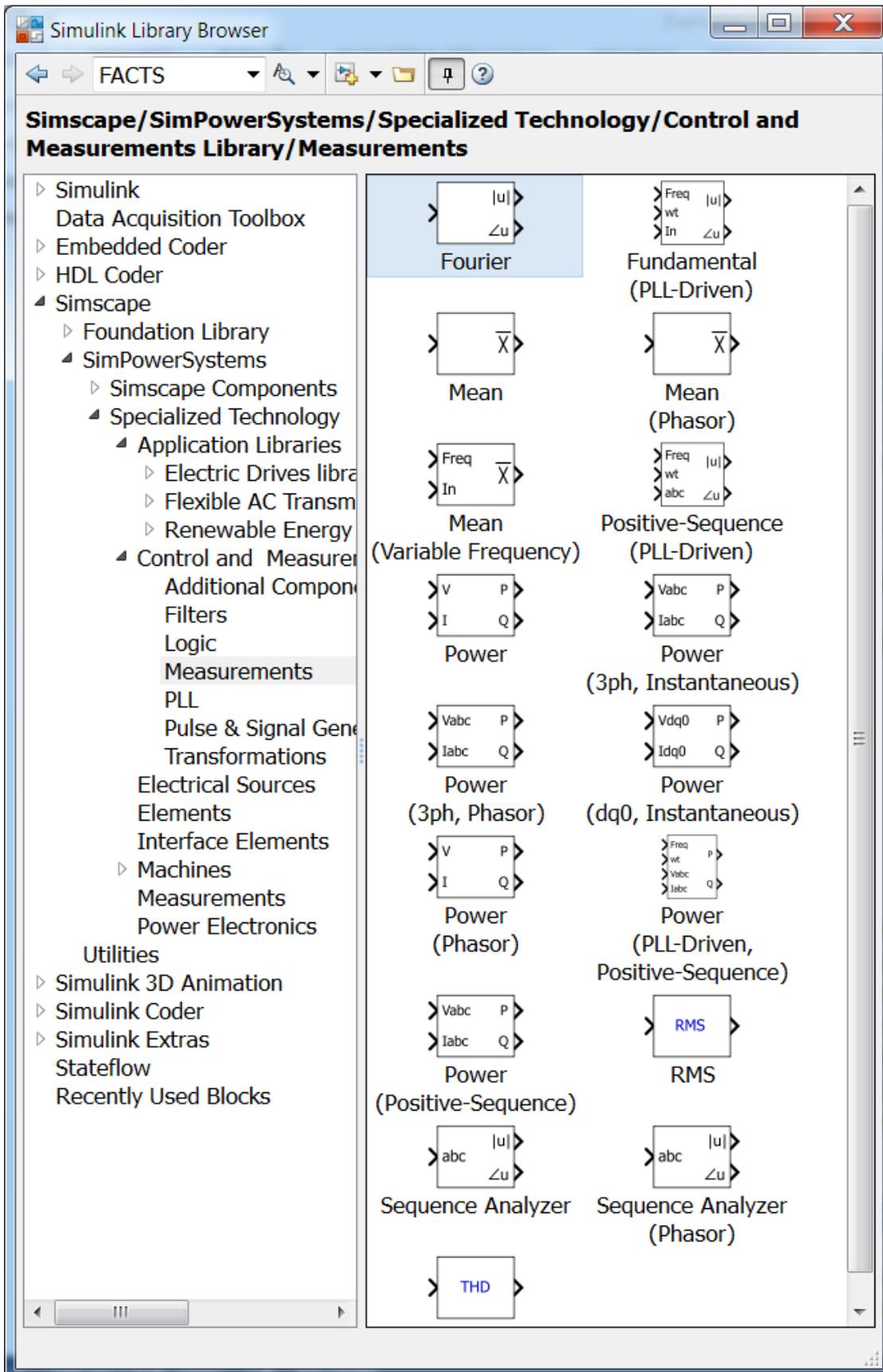


Рис. 1.46

- **PLL** (*Phase Locked Loop*) – блоки системы фазовой автоподстройки частоты (ФАПЧ);
- **Pulse & Signal Generators** – блоки генераторов импульсов и сигналов;
- **Transformations** – блоки преобразователей координат.

3) **Electrical Sources** – источники электрической энергии и сигналов (Рис. 1.47):

- **AC Current Source** – блок источника переменного тока;
- **AC Voltage Source** – блок источника переменного напряжения;
- **Battery** – блок обобщенной модели батареи;
- **Controlled Current Source** – блок управляемого источника тока;
- **Controlled Voltage Source** – блок управляемого источника напряжения;
- **DC Voltage Source** – блок источника постоянного напряжения;
- **Three-Phase Programmable Voltage Source** – блок трехфазного программируемого источника напряжения;
- **Three-Phase Source** – блок трехфазного источника.

4) **Elements** – линейные и нелинейные компоненты электротехнических и электронных устройств [2]:

- **Breaker** – выключатель, отключающий цепь при прохождении кривой тока через нулевое значение (рис. 1.48);
- **Connection Port** – порт соединения;
- **Distributed Parameters Line** – линия с распределенными параметрами;
- **Ground** – заземление;
- **Grounding Transformer** – заземляющий трансформатор;
- **Linear Transformer** – трансформатор без учета насыщения;
- **Multi-Winding Transformer** – многообмоточный трансформатор;
- **Mutual Inductance** – взаимная индуктивность;
- **Neutral** – блок служит для создания общих узлов между блоками;
- **Parallel RLC Branch** – однофазная параллельная RLC цепь;
- **Parallel RLC Load** – параллельная RLC нагрузка;
- **Pi Section Line** – линия электропередачи с сосредоточенными параметрами;

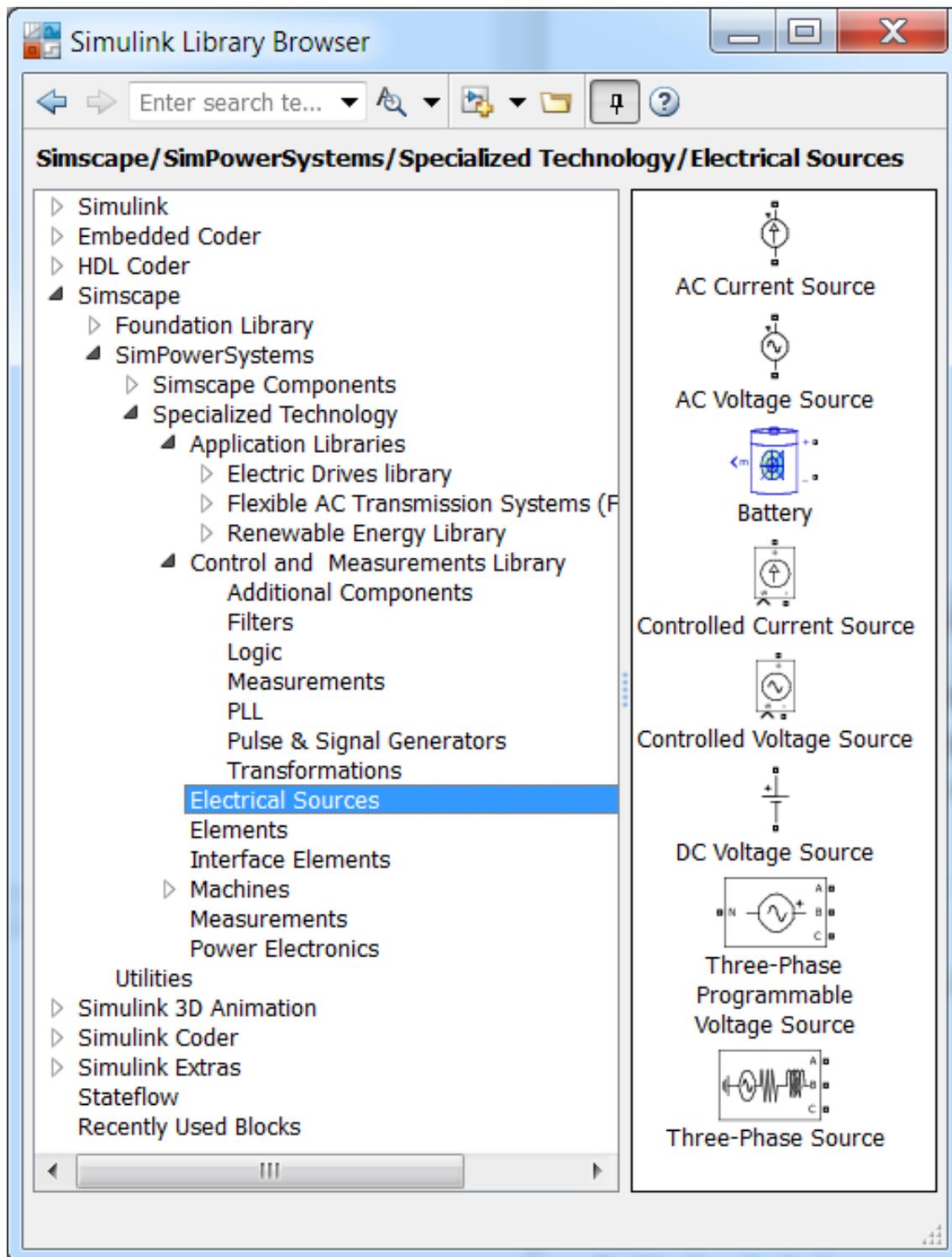


Рис. 1.47

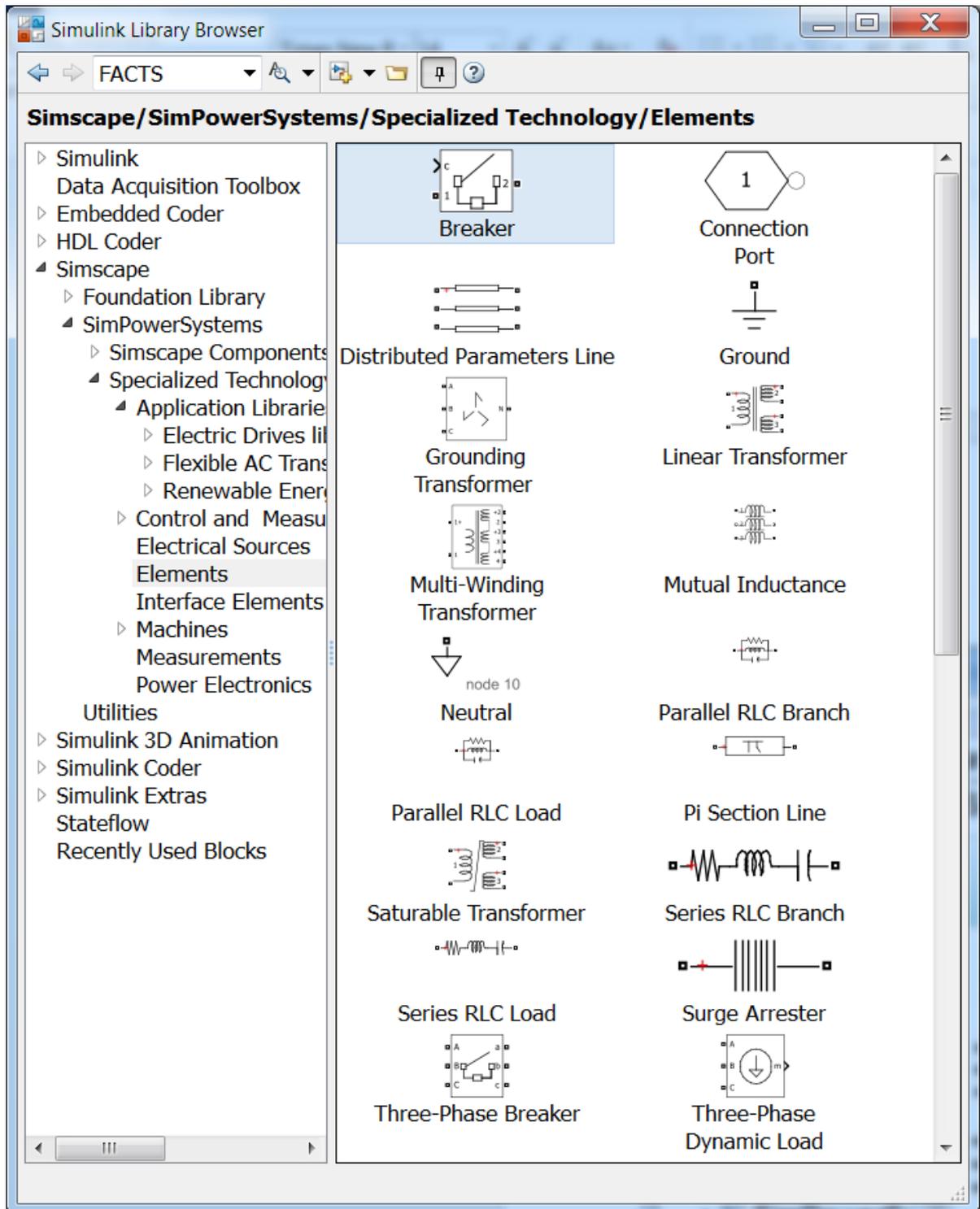


Рис. 1.48

- **Series RLC Branch** – последовательная RLC цепь;
- **Series RLC Load** – последовательная RLC нагрузка;
- **Surge Arrester** – разрядник для защиты от перенапряжений;
- **Three-Phase Breaker** – трехфазный выключатель, отключающий токи фаз при прохождении их кривых через нулевые значения;
- **Three-Phase Fault** – блок трехфазных повреждений, который позволяет моделировать различные замыкания (Рис. 1.49);
- **Three-Phase Harmonic Filter** – трехфазный фильтр гармоник;
- **Three-Phase Mutual Inductance Z1-Z0** – трехфазная взаимная индуктивность;
- **Three-Phase PI Section Line** – трехфазная линия электропередачи с сосредоточенными параметрами;
- **Three-Phase Parallel RLC Branch** – трехфазная параллельная RLC цепь;
- **Three-Phase Parallel RLC Load** – трехфазная параллельная RLC нагрузка;
- **Three-Phase Series RLC Branch** – трехфазная последовательная RLC цепь;
- **Three-Phase Series RLC Load** – трехфазная последовательная RLC нагрузка;
- **Three-Phase Transformer (Three Windings)** – трехобмоточный трехфазный трансформатор;
- **Three-Phase Transformer (Two Windings)** – блок трехфазного двухобмоточного трансформатора;
- **Three-Phase Transformer 12 Terminals** – трехфазный трансформатор с 12 выводами (клеммами);
- **Three-Phase Transformer Inductance Matrix Type (Three Windings)** – блок трехфазного трехобмоточного трансформатора (автотрансформатора) со стержневой (броневой) конструкцией магнитопровода;
- **Three-Phase Transformer Inductance Matrix Type (Two Windings)** – блок трехфазного двухобмоточного трансформатора (автотрансформатора) со стержневой (броневой) конструкцией магнитопровода;
- **Zigzag Phase-Shifting Transformer** – трансформатор с соединением "зигзаг".

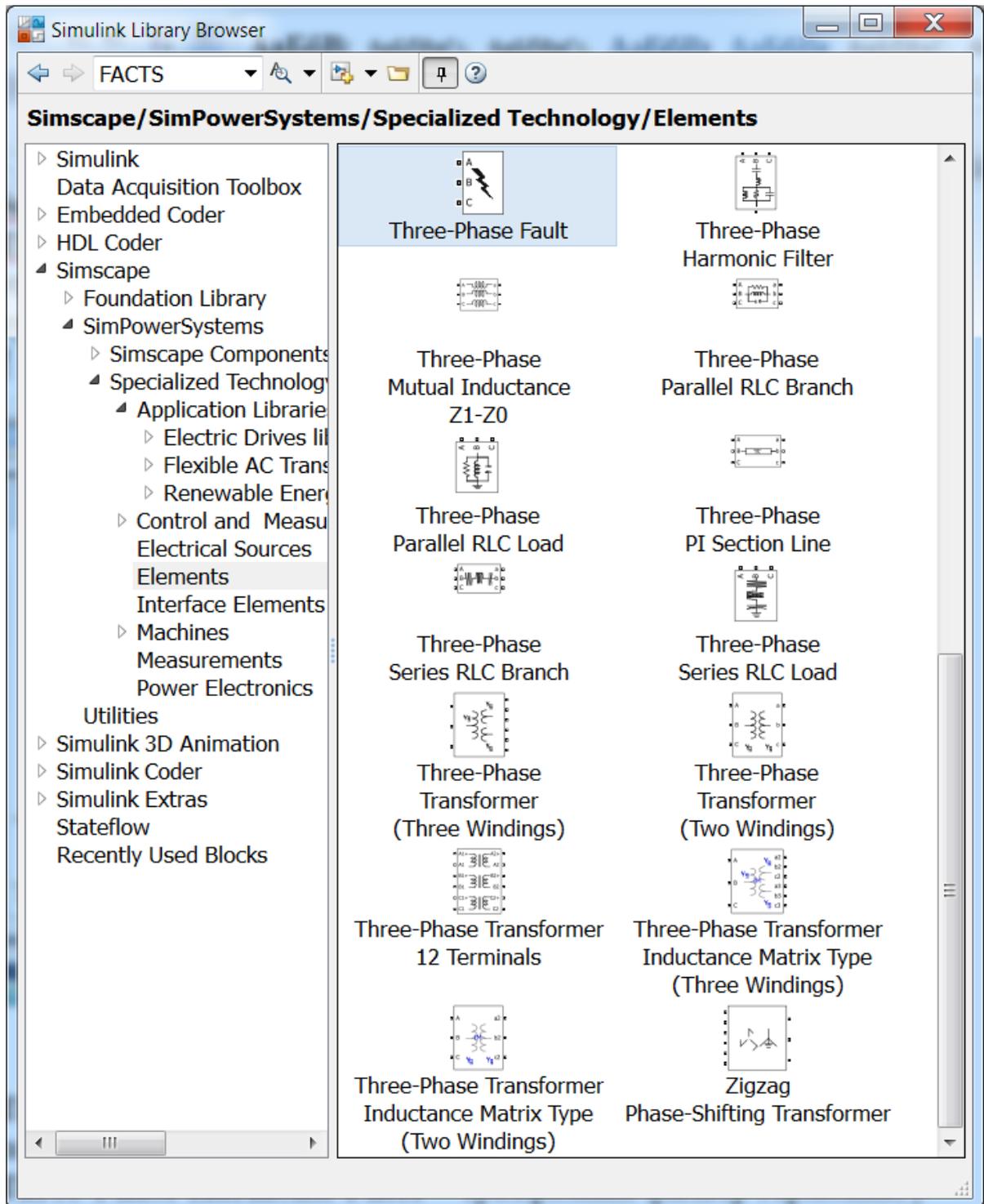


Рис. 1.49

5) **Interface Elements** – элементы интерфейса между **SimPowerSystems** и электрическими цепями расширения **Simscape** [10]:

- **Current-Voltage Simscape Interface** – ампер-вольтный интерфейс, который позволяет соединять цепи **SimPowerSystems** и электрические цепи **Simscape** (Рис. 1.50);
- **Current-Voltage Simscape Interface (gnd)** – ампер-вольтный интерфейс, который позволяет соединять **SimPowerSystems** и электрические цепи **Simscape** с использованием заземления;
- **Voltage-Current Simscape Interface** – вольт-амперный интерфейс, который позволяет соединять цепи **SimPowerSystems** и электрические цепи **Simscape**;
- **Voltage-Current Simscape Interface (gnd)** вольт-амперный интерфейс, который позволяет соединять **SimPowerSystems** и электрические цепи **Simscape** с использованием заземления.

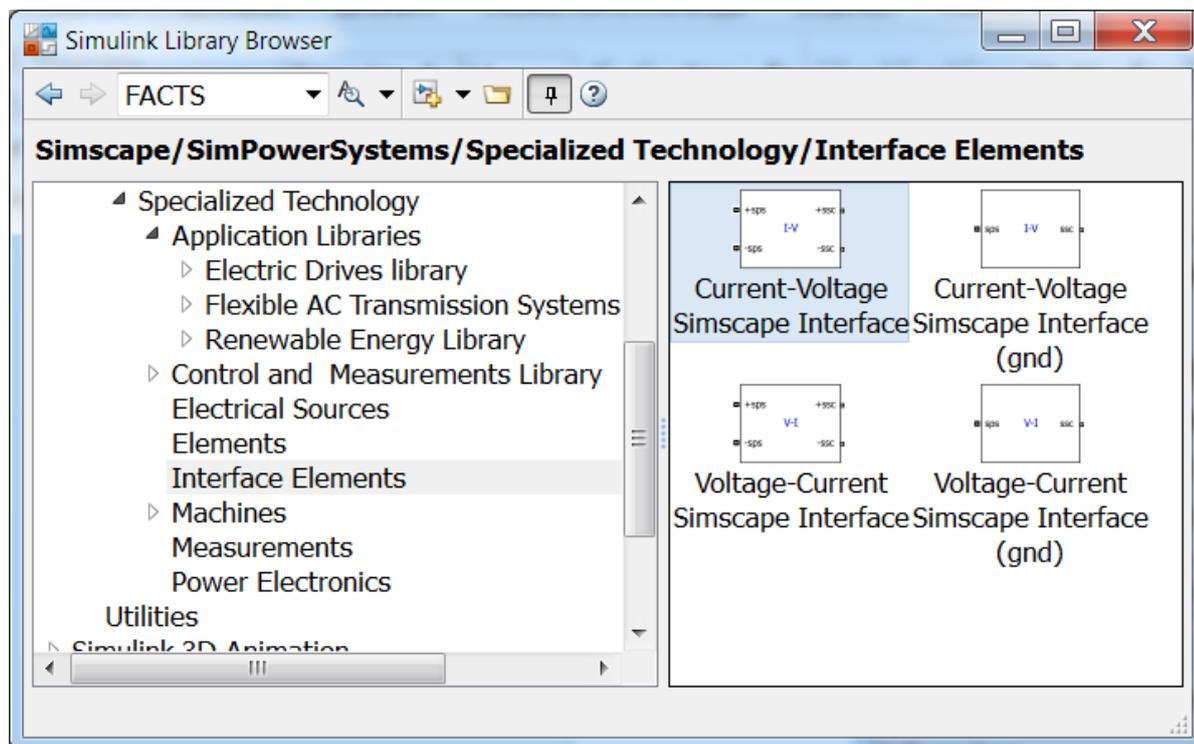


Рис. 1.50

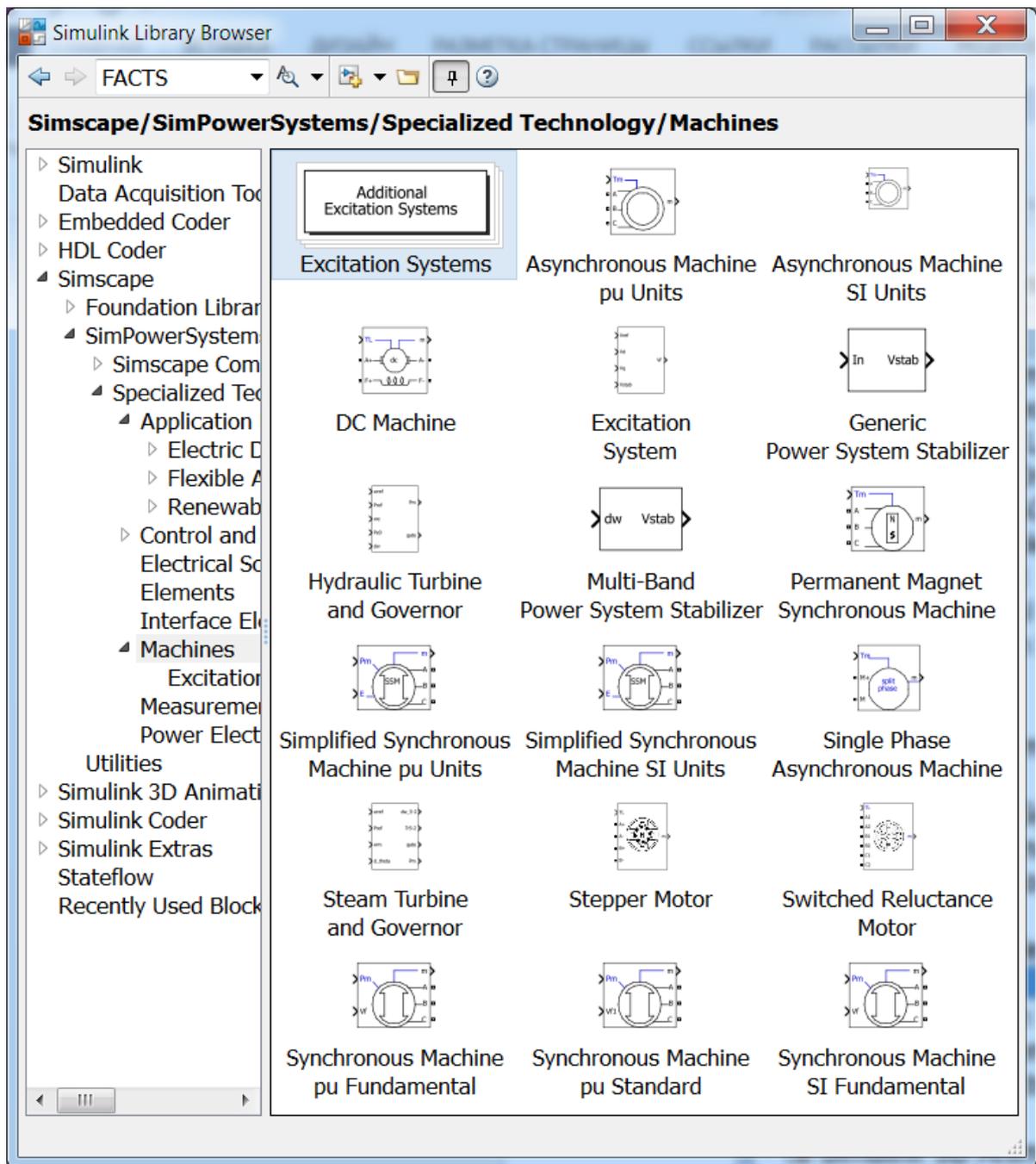


Рис. 1.51

6) **Machines** – блоки для электрических машин (Рис. 1.51):

- **Excitation Systems** – блоки систем возбуждения синхронных машин (папка):
- **Asynchronous Machine SI Units** – блок асинхронной машины с параметрами, заданными в международной системе единиц;
- **Asynchronous Machine pu Units** – блок асинхронной машины с параметрами в относительных единицах;
- **DC Machine** – блок машины постоянного тока;

- **Excitation System** – блок системы возбуждения синхронных машин;
- **Generic Power System Stabilizer** – универсальный стабилизатор энергетической системы;
- **Hydraulic Turbine and Governor** – гидравлическая турбина и регулятор;
- **Multi-Band Power System Stabilizer** – многополосный стабилизатор энергосистемы;
- **Permanent Magnet Synchronous Machine** – синхронная машина с постоянными магнитами;
- **Simplified Synchronous Machine SI Unit** – блок на основе упрощенной модели синхронной машины с параметрами, заданными в международной системе единиц;
- **Simplified Synchronous Machine pu Units** – блок на основе упрощенной модели синхронной машины в относительных единицах;
- **Single Phase Asynchronous Machine** – однофазная асинхронная машина (Рис. 1.16);
- **Steam Turbine and Governor** – паровая турбина и регулятор;
- **Stepper Motor** – шаговый двигатель;
- **Switched Reluctance Motor** – вентильный реактивный двигатель;
- **Synchronous Machine SI Fundamental** – фундаментальная синхронная машина с параметрами в системе СИ;
- **Synchronous Machine pu Fundamental** – фундаментальная синхронная машина с параметрами в относительных единицах;
- **Synchronous Machine pu Standard** – стандартная синхронная машина с параметрами в относительных единицах.

7) **Measurements** – блоки измерений (Рис. 1.52):

- **Current Measurement** – блок измерения тока;
- **Impedance Measurement** – блок измерения полного сопротивления;
- **Load Flow Bus (BUS_1)** – блок шины потока нагрузки (мощности), с помощью которого можно определить потоки нагрузки (мощности) в ветвях, подключенных к шине (узлу). Для получения результатов необходимо двойным щелчком по блоку **powergui** вызвать его окно и нажать кнопку **Load Flow**. Появится окно **Power Load Flow Tool** с данными системы (сети).

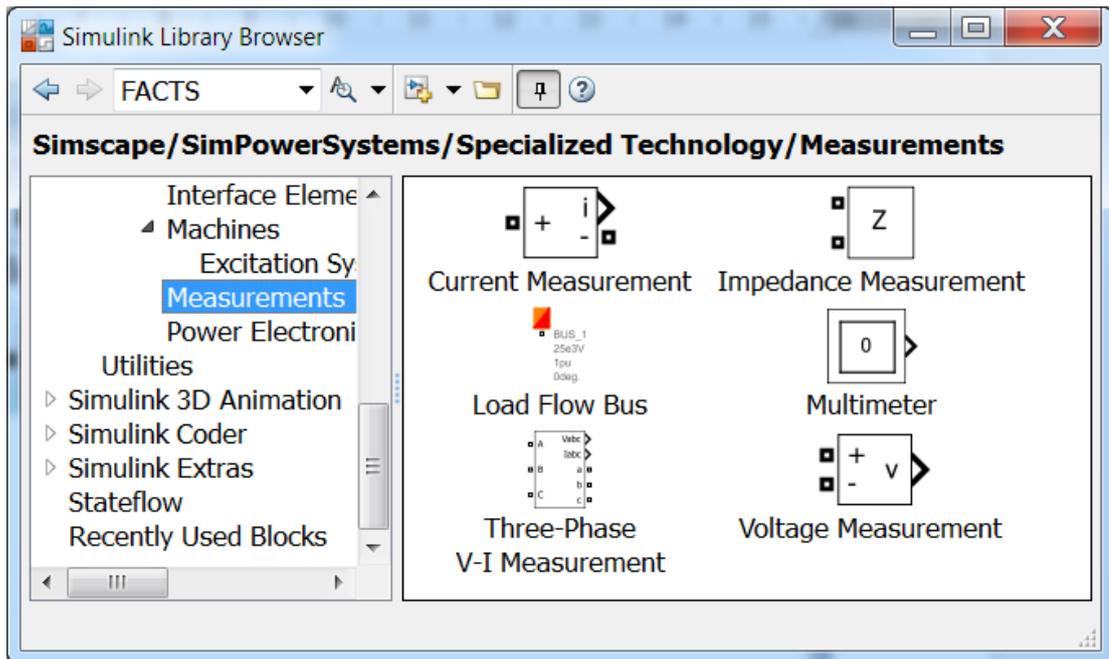


Рис. 1.52

После нажатия кнопки **Compute** получим численные значения потоков мощности в соответствующих ветвях, отходящих от шин (узлов). При необходимости, изменяем параметры системы, возвращаемся в окно **Power Load Flow Tool**, нажимаем кнопки **Update**, **Compute** и получаем новый результат. Следует отметить, что программу **powergui** можно вызвать и из окна команд **MATLAB**, как и пример **power_turbine** использования блока **Load Flow Bus**.

- **Multimeter** – блок универсального измерительного прибора;
- **Three-Phase V-I Measurement** – блок измерения напряжения и тока трехфазной системы;
- **Voltage Measurement** – блок измерения напряжения.

8) **Power Electronics** – блоки устройств силовой электроники (Рис. 1.53):

- **Detailed Thyristor** – блок точной модели тиристора;
- **Diode** – блок диода;
- **Gto** – блок полностью управляемого тиристора;
- **Ideal Switch** – блок идеального ключа;
- **IGBT** – блок биполярного транзистора с изолированным затвором;
- **IGBT/Diode** – блок биполярного транзистора с изолированным затвором и шунтированного обратным диодом;
- **Mosfet** – блок силового полевого транзистора с параллельно включенным обратным диодом;

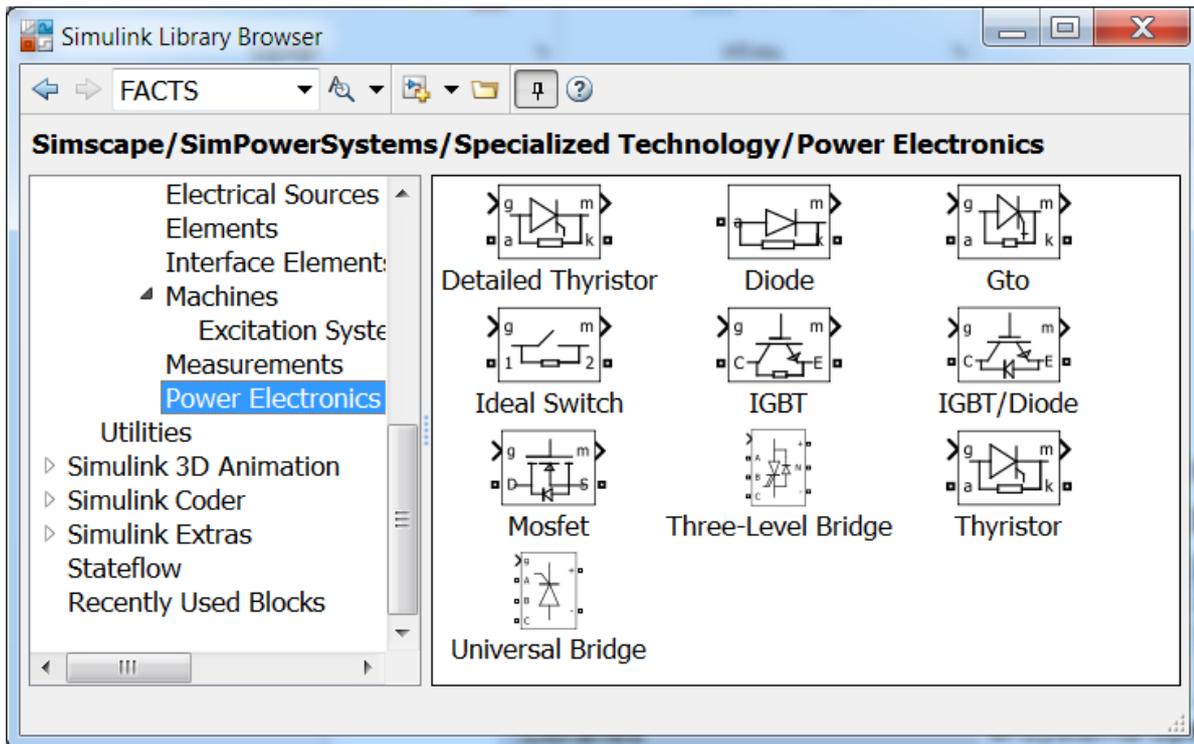


Рис. 1.53

- **Three-Level Bridge** – блок трехуровневого моста;
- **Thyristor** – блок упрощенной модели тиристора;
- **Universal Bridge** – блок универсального моста.

9) **Powergui** – графический интерфейс пользователя расширения **SimPowerSystems**.

2. МОДЕЛИРОВАНИЕ ЭЛЕМЕНТОВ, УСТРОЙСТВ И СИСТЕМ ЭЛЕКТРОЭНЕРГЕТИКИ

2.1. Моделирование переходных процессов в индуктивности, конденсаторе и выпрямителе

Моделирование переходных процессов с использованием возможностей **SimPowerSystems** сводится к нахождению необходимых блоков, их соединению, настройке решателя и выводу результатов.

Рассмотрим следующий пример.

Пример 2.1.

Необходимо рассчитать переходной процесс при подключении к источнику постоянного напряжения через активное сопротивление катушки индуктивности и переходной процесс при аналогичном подключении конденсатора.

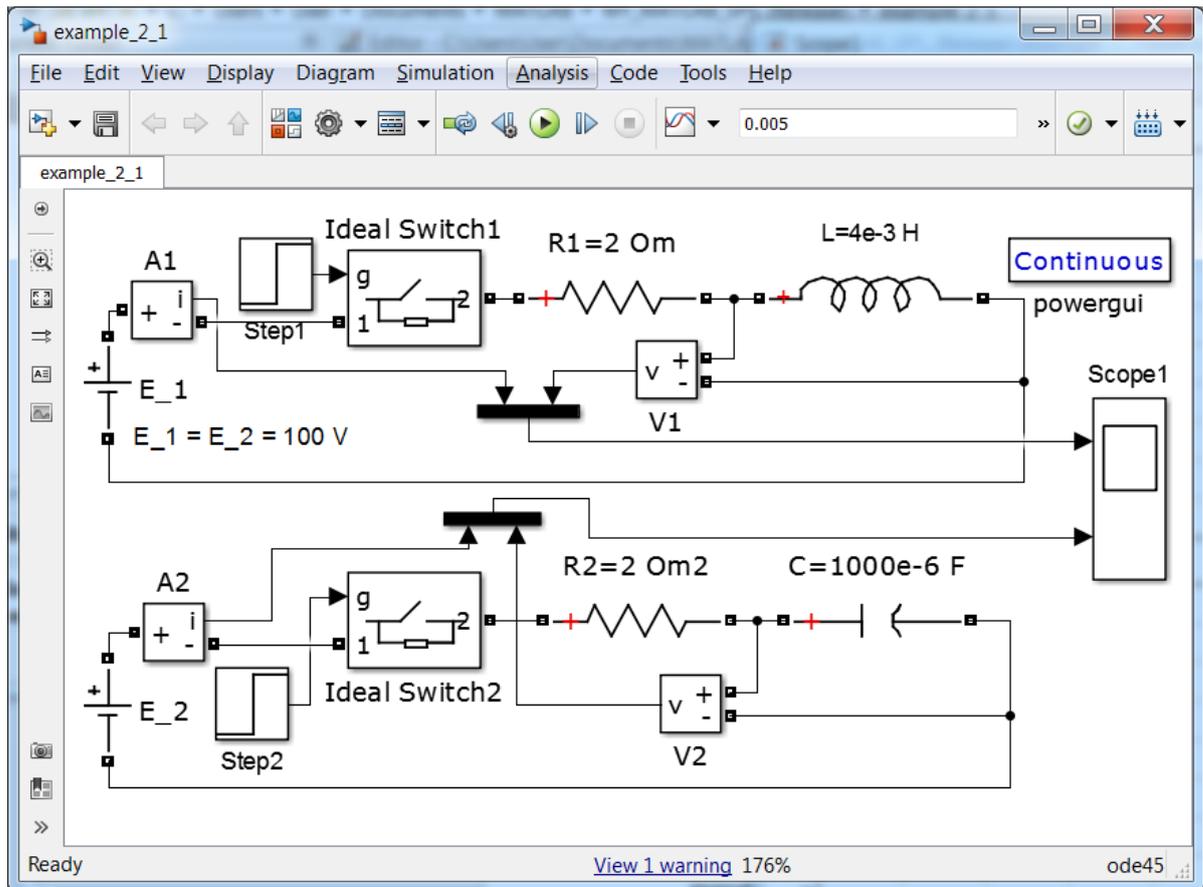


Рис. 2.1

Параметры источников ЭДС, активных сопротивлений, катушки индуктивности и конденсатора имеют следующие значения: $E_1 = E_2 = 100\text{В}$; $R_1 = R_2 = 2\text{Ом}$; $L = 4\text{мГн}$; $C = 1000\text{мкФ}$.

Решение

Создаем новую модель (**example_2_1.**) В окне модели перетаскиваем из библиотеки следующие блоки (Рис. 2.1): **DC Voltage Source** (2 шт.), **Current Measurement** (2 шт.), **Ideal Switch** (2 шт.), **Step** (2 шт.), **Series RLC Branch** (4 шт.), **Voltage Measurement** (2 шт.), **Mux** (2 шт.), **Scope**. Соединяем блоки согласно схеме модели.

Изменяем подписи блоков. Наводим на неё курсор, один раз щелкаем левую кнопку мыши и вводим текст на латинице, например, E_1 .

В окна параметров блоков E_1 и E_2 (**DC Voltage Source**) записываем напряжение 100В . В блоки **Step_1** и **Step_2** записываем ноль. В блоках **R_1**, **R_2**, **L**, **C** (**Series RLC Branch**) выбираем необходимый тип ветви и вводим соответствующие значения параметров (Рис. 2.2). Заносим время расчета: $0,005\text{с}$.

После нажатия кнопки **Run** получим кривые, представленные на рис 2.3. Здесь, пунктиром показаны токи, сплошной линией – напряжения. На верхнем графике показаны кривые тока и напряжения на катушке индуктивности, а на нижнем – кривые тока и напряжения на конденсаторе.

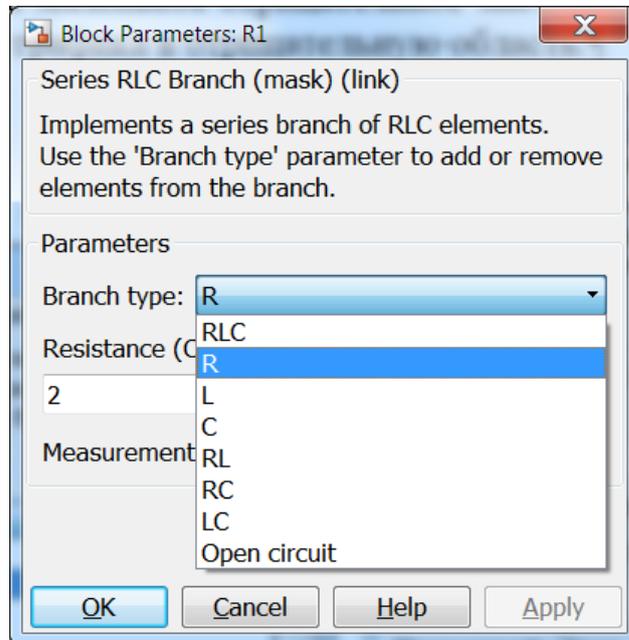


Рис. 2.2

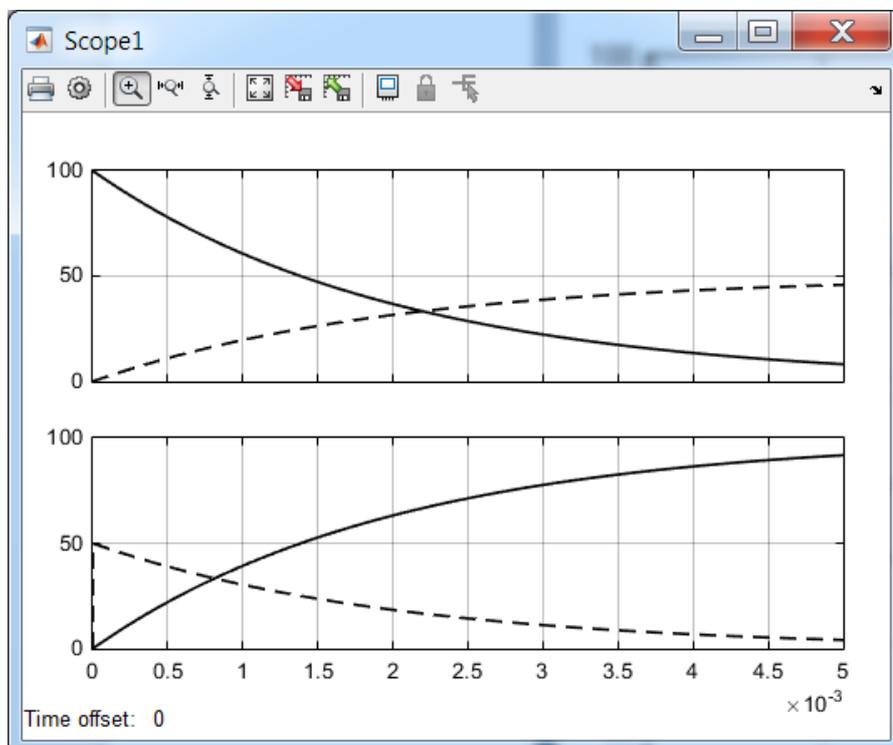


Рис. 2.3

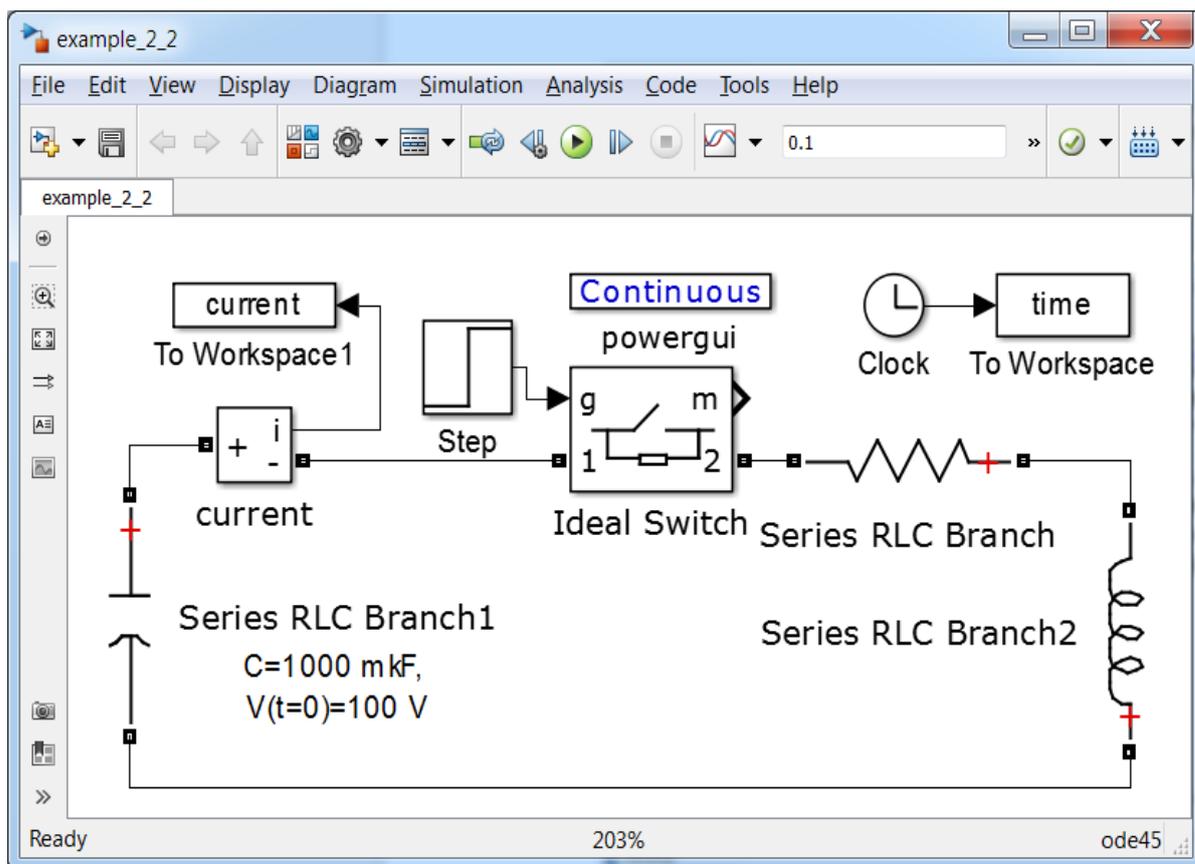


Рис. 2.4

При моделировании часто возникает задача изменения в цикле какого-либо параметра с целью анализа его влияния на объект исследований. Такая задача легко решается с помощью **SimPowerSystems** модели, которая запускается программой, написанной на **MATLAB**.

Пример 2.2.

Определить значение активного сопротивления, при котором периодический характер разряда конденсатора на RL цепь становится аperiodическим. Параметры цепи: $C = 10000 \text{ мкФ}$, $L = 100 \text{ мГн}$.

Конденсатор предварительно заряжен до 100 В . Время расчета $0,1 \text{ с}$. Активное сопротивление R изменяется от 5 до 25 Ом с шагом 5 Ом .

Решение

Для решения этой задачи воспользуемся блоками **SimPowerSystems** и создадим модель, представленную на рис. 2.4 (**example_2_2**). С помощью двух блоков **To Workspace** данные с измерителя тока и часов передаются в рабочую область и помещаются в переменные **current** и **time**.

Для организации цикла воспользуемся М-файлом (**Script**), текст которого представлен ниже.

```

% example_2_2_m.m
clear all; % Очистка Рабочей области (Workspace)
open('example_2_2.slx') % Открытие модели figure;
% Подготовка окна для вывода кривых графика
xlabel('time, s'); ylabel('current, A'); %Подписи осей
grid on; % Создание координатной сетки
for R = 5:5:26 % Цикл расчета
% Присвоение значения активному сопротивлению блока:
set_param('example_2_2/Series RLC
Branch', 'Resistance', num2str(R));
sim('example_2_2.slx'); % Запуск модели
switch R % Использование конструкции выбора switch для
% изменения цвета кривых
case 5
    C='g'; % Зеленый цвет
case 10
    C='r'; % Красный
case 15
    C='b'; % Синий
case 20
    C='m'; % Magenta -- малиновый
case 25
    C='k'; % Black - k - черный (т.к. b закреплен за
    % синим)
end %Конец switch
line(time, current, 'Color', C, 'LineWidth', 2); % Построение
% графиков
end % Окончание цикла

```

Описание этого m-файла не дается, поскольку комментарии, данные к каждой строке m-файла, позволяют раскрыть алгоритм его работы.

Результаты расчета, представленные на рис. 2.5, свидетельствуют о том, что при активном сопротивлении $R \geq 200\text{Ом}$ разряд конденсатора принимает апериодический характер.

Следует отметить, что основной целью данного примера была демонстрация взаимодействия m-файла (**Script**) с моделью **SimPowerSystems**, а не определение точного численного значения активного сопротивления, которое может быть рассчитано аналитически.

Особую актуальность **SimPowerSystems** приобретает при решении нелинейных задач, когда аналитические методы становятся малоэффективными из-за их громоздкости или низкой точности, вызванной допущениями. В этом случае для решения системы дифференциальных уравнений этих нелинейных устройств и объектов используют численные методы, большинство из которых реализованы в **MATLAB**.

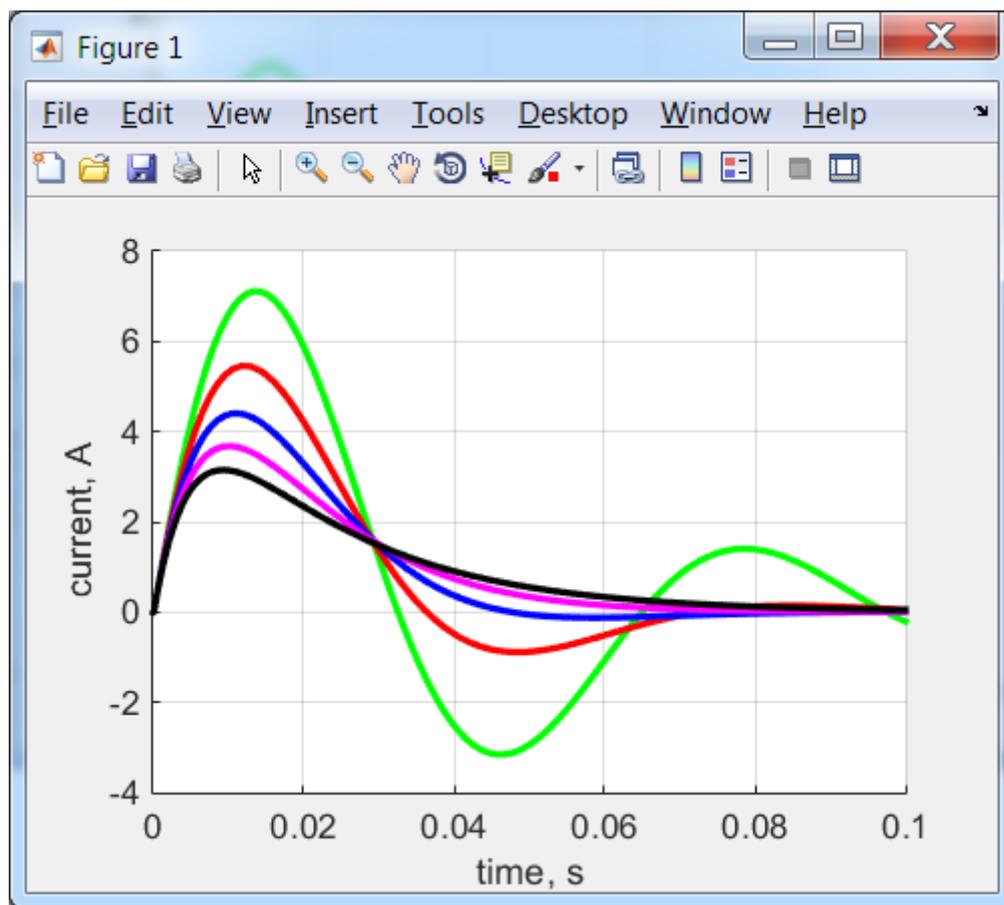


Рис. 2.5

В качестве примера расчета такой нелинейной системы рассмотрим модель однополупериодного выпрямителя.

Пример 2.3.

Для вызова примера набираем в окне команд (**Command Window**) название файла >> **power_diode**. Как всегда, имеются и альтернативные пути: в библиотеке **SimPowerSystems Specialized Technology** переходим в раздел **Power Electronics**, выделяем блок **Diode**, нажимаем правую клавишу мышки, в выпадающем окне переходим на позицию **Help for the Diode block** (Рис. 2.6).

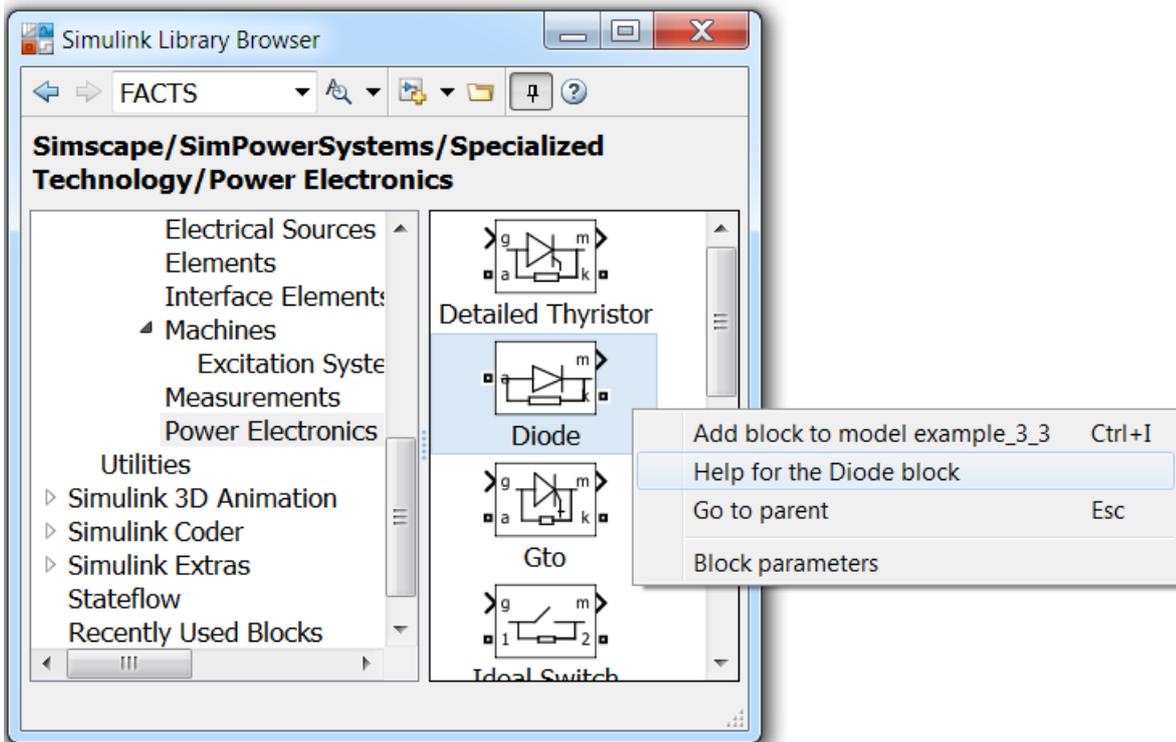


Рис. 2.6

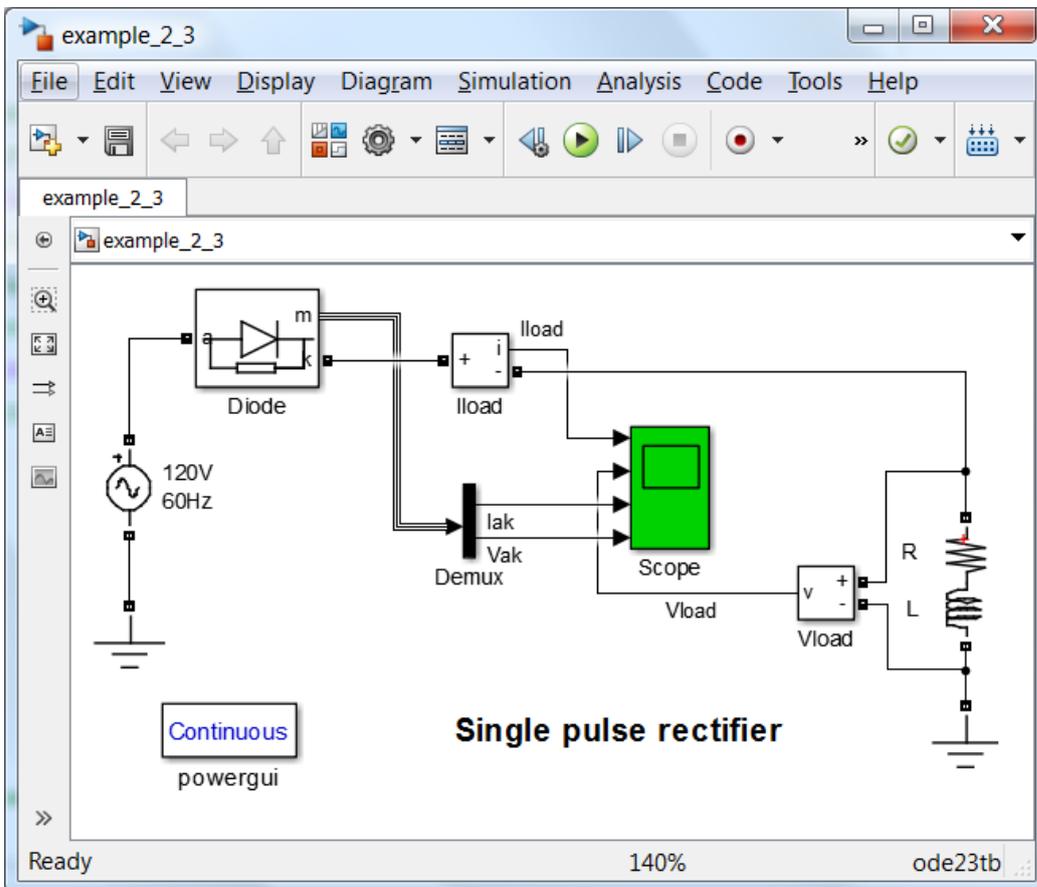


Рис. 2.7

Попадаем в окно с описанием диода, в котором находим заголовок **Example**, под которым нажимаем на выделенное слово **power_diode**.

В папку **power_electronics** с демонстрационными примерами можно попасть и указав следующий путь в окне программы **Компьютер** операционной системы **Windows**: C:\Program Files\MATLAB\R2014b\toolbox\physmod\powersys\powerdemo\power_electronics.

После загрузки сохраняем её в удобной для работы папке под новым именем, например, **example_2_3**. (Если по каким-либо причинам не удалось обнаружить эту модель, то её можно легко собрать, воспользовавшись библиотекой блоков. Параметры блоков имеют следующие значения: у источника они указаны в названии блока, т.е. напряжение – 100 В, частота – 60 Гц; диод может иметь параметры, установленные по умолчанию; в блоке сопротивлений – 1 Ом и 1 мГн.).

В этой модели (рис. 2.7) используются знакомые блоки, однако имеется и новшество: от порта (выхода) **m** блока **Diode** до **Demux** используется "странная" линия. Это объясняется тем, что в окне **Block Parameters Diode** активизирована позиция **Show measurements: Diode** и значение тока и напряжения диода передаются с этого порта на любой приемник расширения **Simulink**.

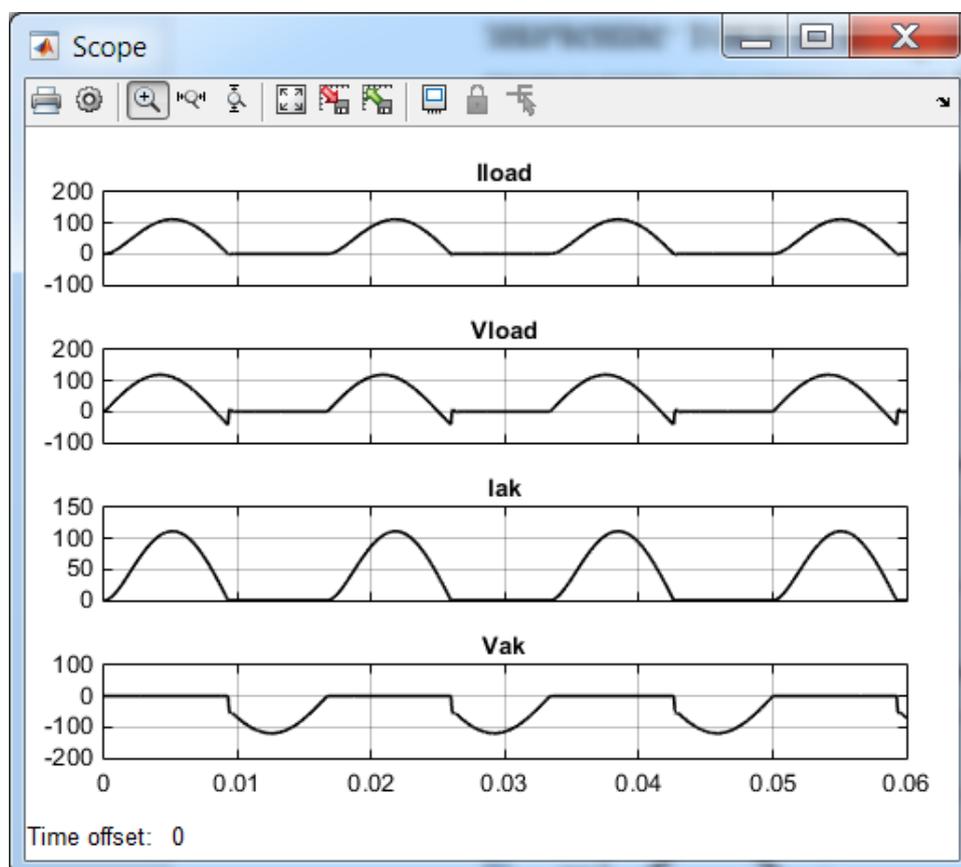


Рис. 2.8

Следует отметить, что источник переменного напряжения имеет частоту 60 Гц . В результате моделирования получаем кривые тока и напряжения на нагрузке и диоде (Рис. 2.8). В данной схеме ток диода равен току нагрузки. При положительном напряжении диод открыт и проводит ток. Падение напряжения на нем незначительное. При отрицательном напряжении и прохождении тока через нулевое значение диод закрывается, его сопротивление резко возрастает и падение напряжения на диоде становится практически равным напряжению сети (V_{ak}).

Следует подчеркнуть, что из-за индуктивности в цепи нагрузки напряжение на нагрузке в конце полупериода принимает отрицательное значение. Известно, что это связано с отставанием тока от напряжения. Когда ток становится равен нулю, то запирающие свойства диода восстанавливаются при наличии на нем отрицательного напряжения.

Нелинейные сопротивления в электрических сетях приводят к появлению высших гармоник, которые оказывают негативное влияние на релейную защиту, электрические машины и другие устройства. В связи с этим, определенный интерес представляет расчет высших гармоник.

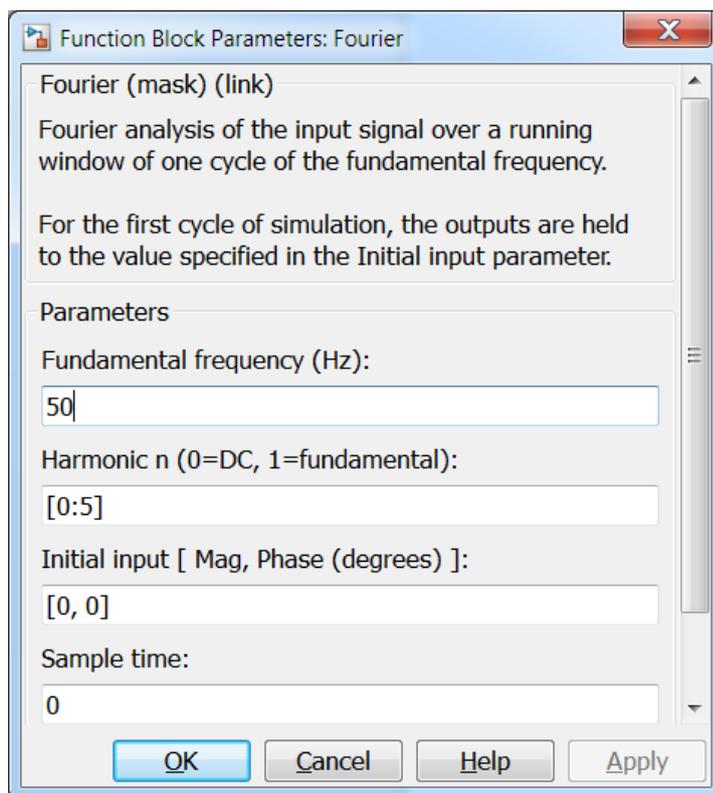


Рис. 2.9

Пример 2.4.

Определим спектральный состав и коэффициент нелинейных искажений (THD) однополупериодного выпрямителя при чисто активной нагрузке и частоте 50 Гц.

Решение

За основу возьмем предыдущий пример, сохранив его как **example_2_4**. В названии блока источника переменного напряжения поменяем цифру **6** на **5**. В окне параметров блока установим частоту **50 Гц**. Удаляем в названии блока сопротивления букву **L**. В окне параметров блока последовательных сопротивлений установим в **Branch type**: значение **R**.

В окно модели переносим блоки **Fourier** и **THD** из раздела **Measurements** библиотеки **SimPowerSystems Specialized Technology**, а также три блока **Display** из библиотеки **Simulink**. Заносим в блок **THD** частоту 50 Гц, а в блок **Fourier** данные, которые представлены на рис. 2.9. Соединяем эти блоки согласно рис. 2.10, причем входным сигналом для них служит ток, который протекает как в нагрузке, так и в источнике переменного напряжения.

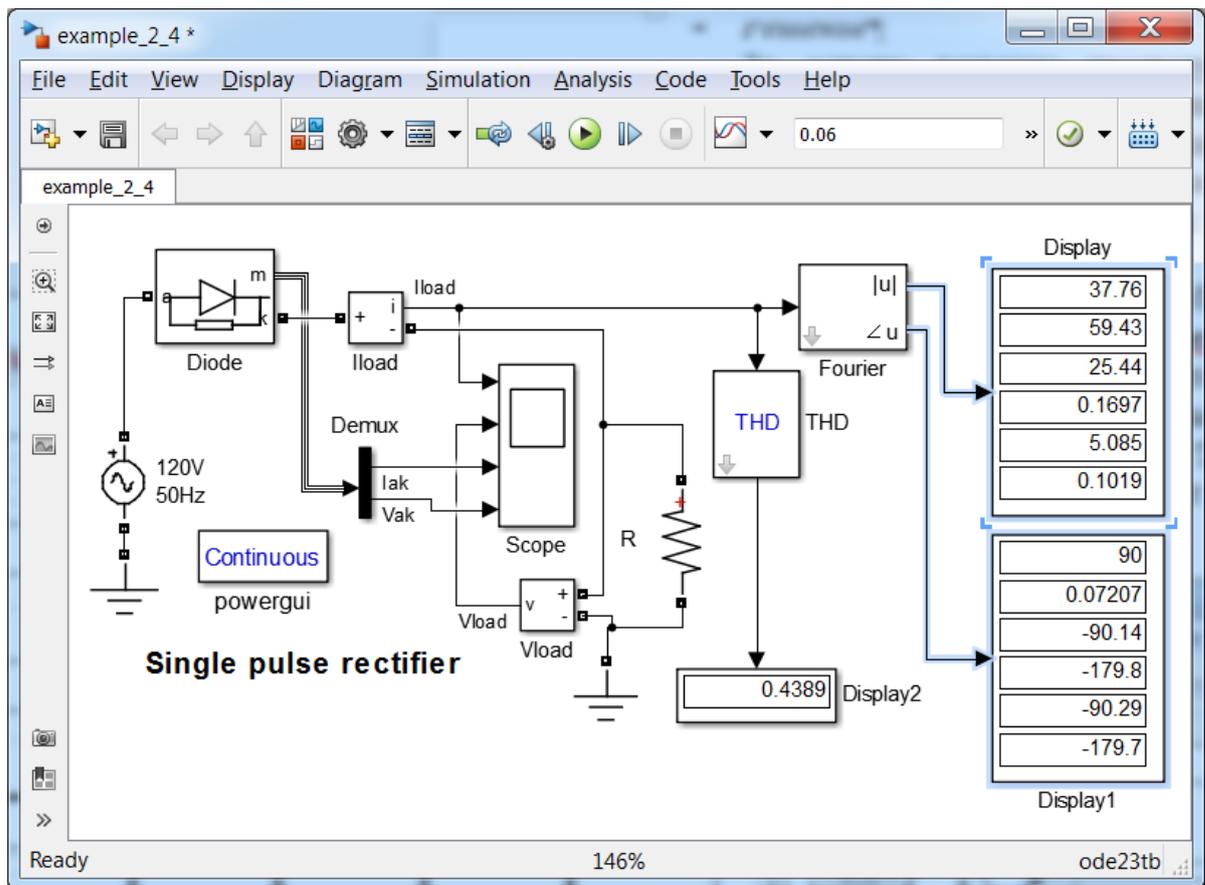


Рис. 2.10

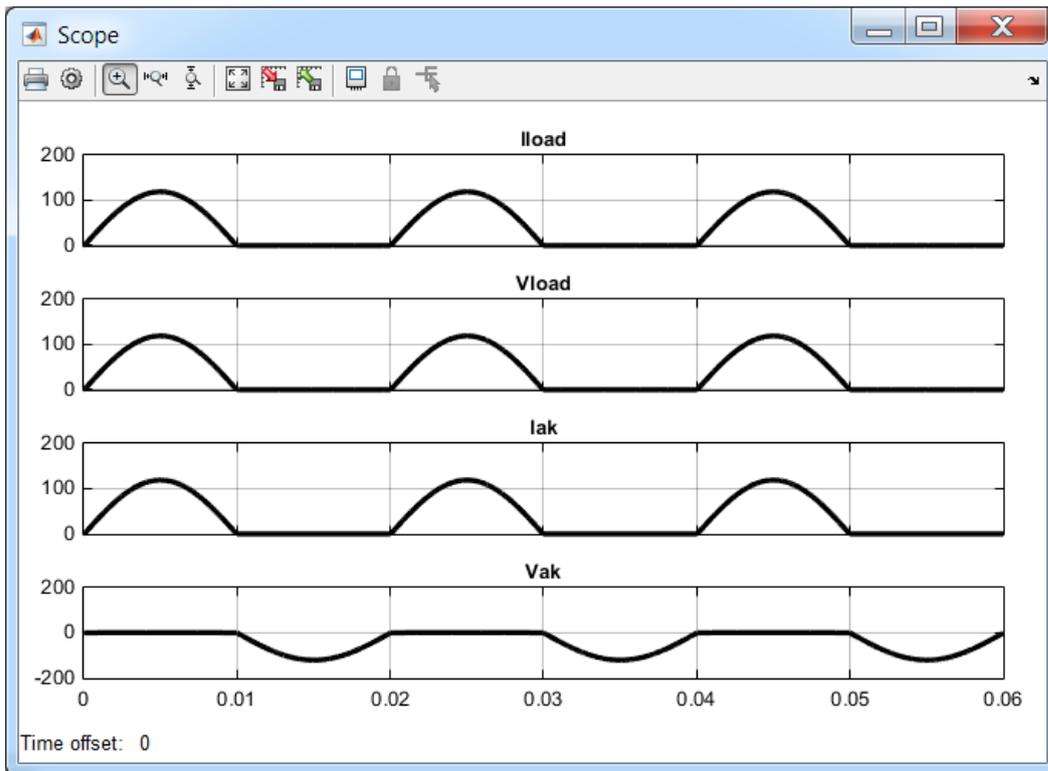


Рис. 2.11

Запускаем программу. На дисплеях отображаются значения амплитуд и фаз гармоник (от нулевой до пятой), а также значение коэффициента нелинейных искажений. Кривые токов и напряжения представлены на рис. 2.11.

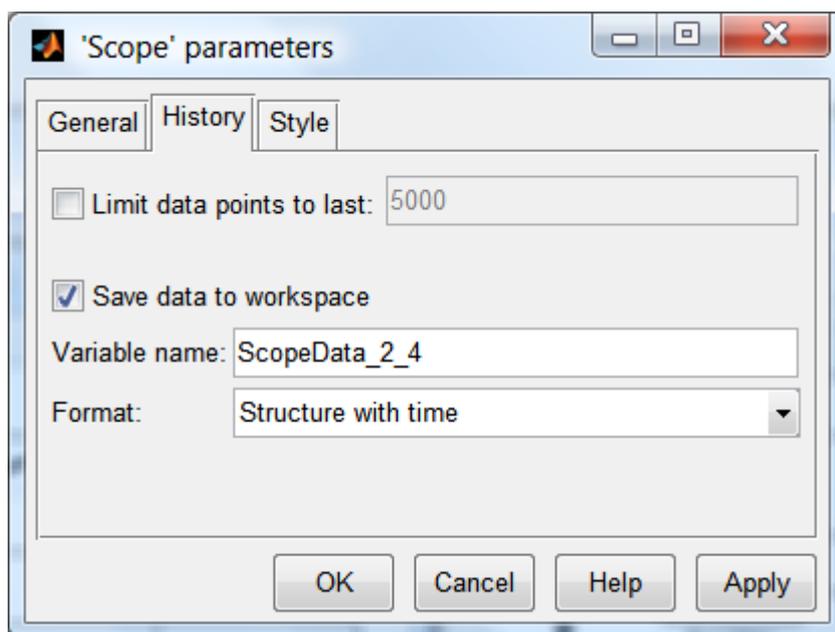


Рис. 2.12

Спектральный состав и коэффициент нелинейных искажений так же можно рассчитать с помощью программы **powergui**. Для этого необходимо записать в файл данные, поступающие в **Scope**. С этой целью в окне **Scope** нажимаем кнопку **Parameters**, переходим во вкладку **History** и вносим в неё изменения согласно рис. 2.12.

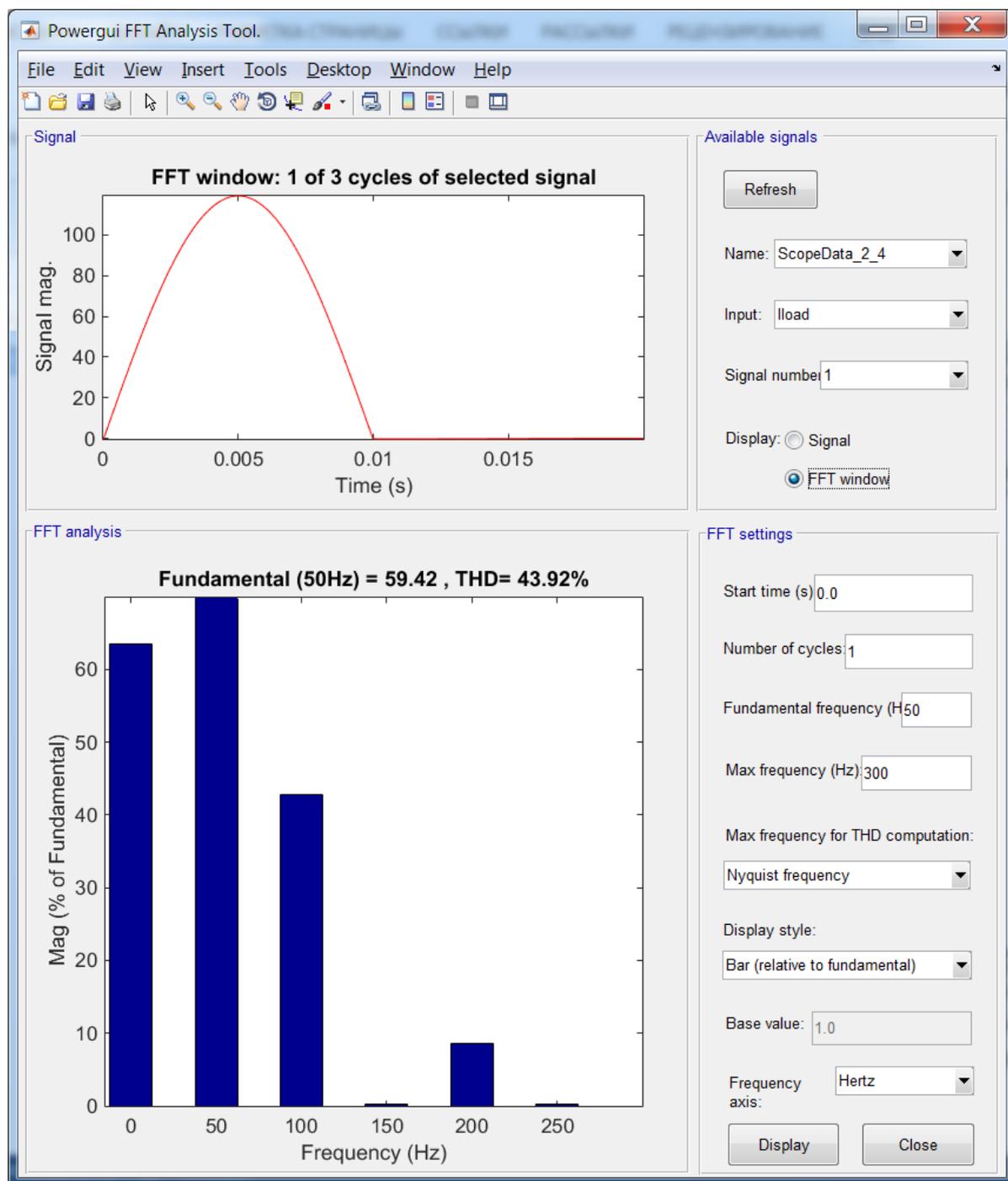


Рис. 2.13

После этого двойным щелчком вызываем программу **powergui**, нажимаем **FFT Analysis (Fast Fourier transform)**, при этом появляется окно, представленное на рис. 2.13.

В верхнем правом углу окна имеются окошки с названием переменной, созданной в **Scope**, и наименованием сигнала, частотный спектр которого требуется найти. Обязательно следует изменить основную частоту (**Fundamental frequency**) на 50 Гц . Как и в предыдущем случае учитываем только шесть гармоник. В связи с этим, максимальную частоту (**Max frequency**) уменьшим до 300 Гц .

В верхнем левом углу имеется окно с одним периодом исследуемой кривой. После нажатия кнопки **Display**, расположенной в нижнем правом углу, осуществляется расчет и появляется спектральный состав сигнала относительно основной (фундаментальной) гармоники. Следует напомнить, что частота основной гармоники составляет 50 Гц и её величина, принятая за 100% , выходит за границы данного графика, т.е. она не равна 70% как это может показаться из графика. Для того, чтобы в этом убедиться, изменим **Display style** (стиль вывода), выбрав **List(relative to fundamental)**. Нажав кнопку **Display**, получим численные значения амплитуд и фаз, представленные на рис. 2.14.

Следует отметить, что эти данные практически совпали с данными, полученными с помощью блоков **Fourier** и **THD** (Рис. 3.10), что свидетельствует об идентичности рассмотренных подходов определения спектрального состава и коэффициента нелинейных искажений (**THD**).

Существенным недостатком однополупериодного выпрямителя является значительный коэффициент нелинейных искажений и наличие постоянной составляющей в токе, протекающем через диод. В связи с этим широкое распространение получил трехфазный мостовой выпрямитель, модель которого рассматривается в следующем примере.

Sampling time	=	1.8388e-05 s	
Samples per cycle	=	1088	
DC component	=	37.75	
Fundamental	=	59.42 peak (42.02 rms)	
THD	=	43.92%	
0 Hz (DC):		63.52%	90.0°
50 Hz (Fnd):		100.00%	0.1°
100 Hz (h2):		42.82%	269.9°
150 Hz (h3):		0.31%	180.3°
200 Hz (h4):		8.56%	269.8°
250 Hz (h5):		0.18%	180.4°

Рис. 2.14

Пример 2.5.

Определим спектральный состав и коэффициент нелинейных искажений (THD) трехфазного мостового выпрямителя, работающего на активную нагрузку, при частоте сети 50 Гц.

Решение

Сохраним модель *примера 2.4* под новым названием **example_2_5**. Скопируем блок источника переменного напряжения для создания фаз **B** и **C**. Откроем окна параметров этих блоков и занесем в них сдвиги фаз **Phase (deg): 120** и **-120** соответственно. Воспользуемся блоком **Three-Phase V-I Measurement** для измерения трехфазных токов и напряжений. Перенесем блоки **Demux** и **Terminator**, которые нужны для ввода значений фазы **A**, в блоки **Fourier** и **THD**. Переносим из библиотеки блок **Universal Bridge** и подключаем его к трехфазному источнику напряжения и нагрузке постоянного тока, активное сопротивление которой составляет 1 Ом. Соединим все блоки согласно схеме модели, представленной на рис. 2.15.

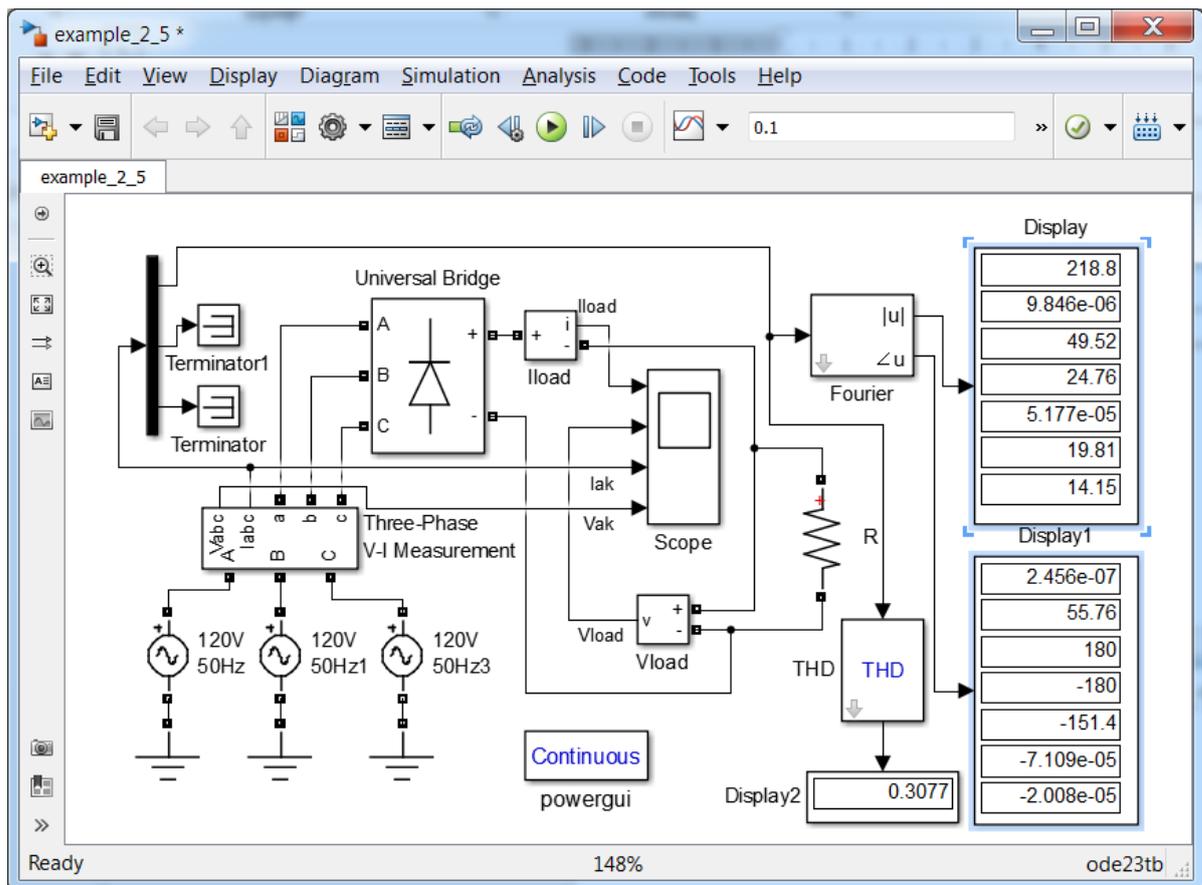


Рис. 2.15

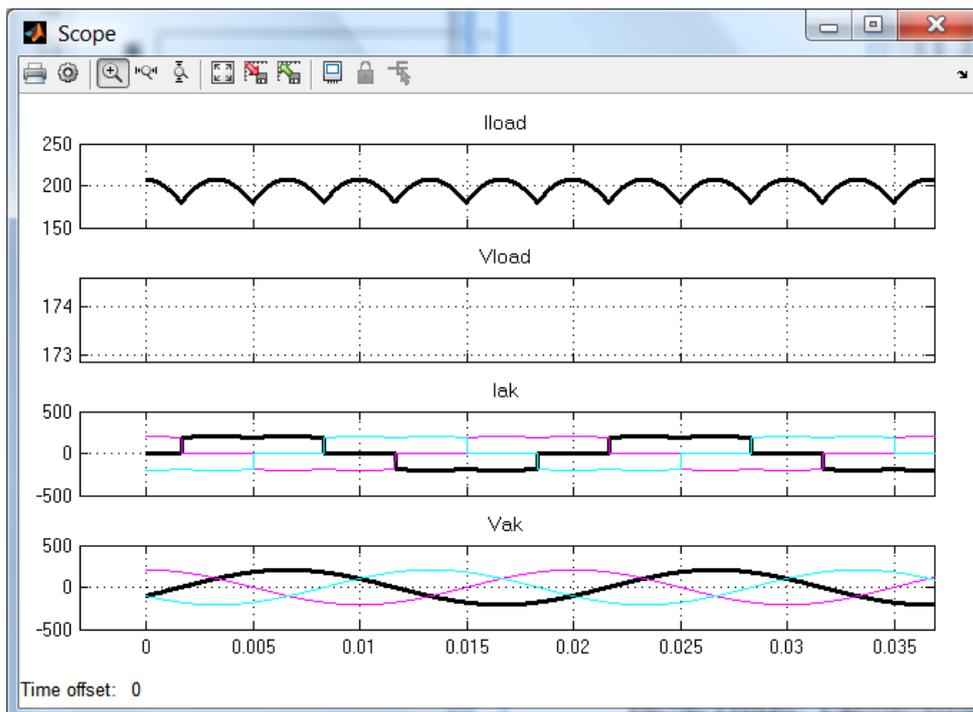


Рис. 2.16

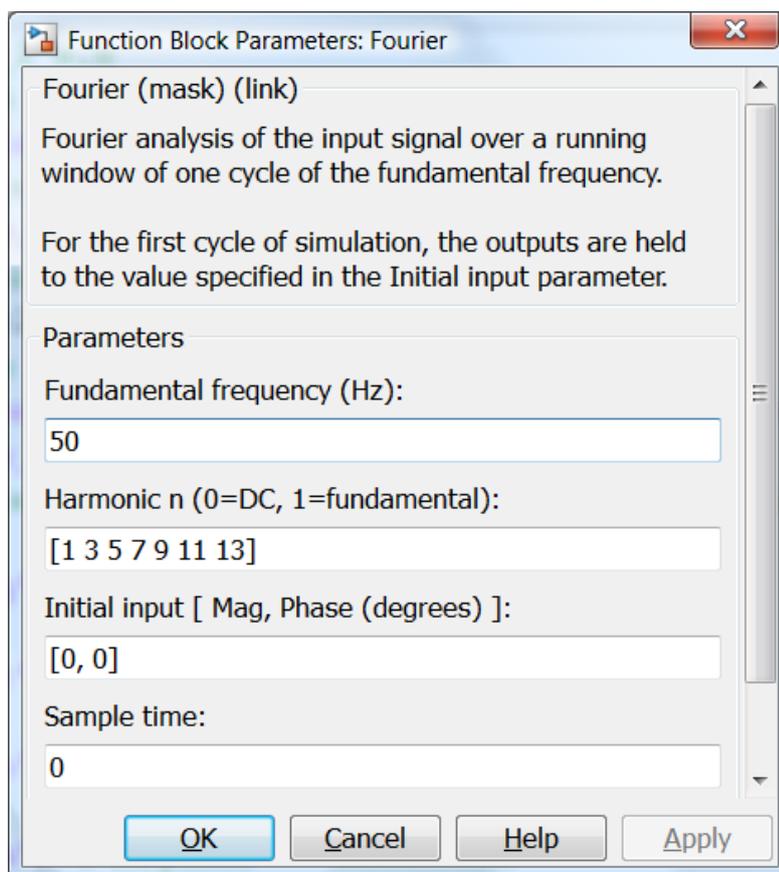


Рис. 2.17

Графические результаты моделирования, представленные на рис. 2.16, свидетельствуют о том, что кривые тока на входе выпрямителя симметричны относительно оси времени, т.е. в них присутствуют только нечетные гармоники.

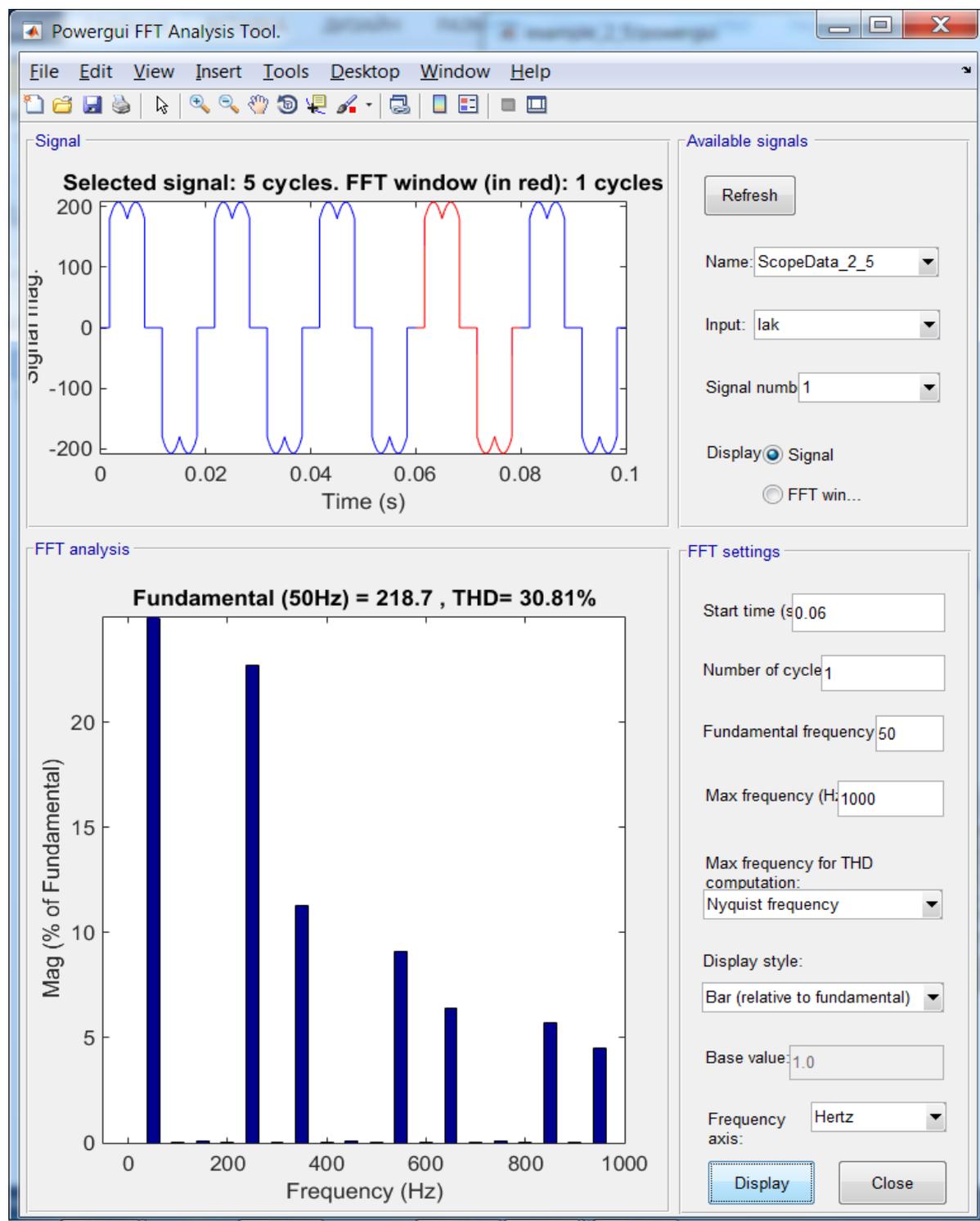


Рис. 2.18

Постоянных составляющих в кривых токов нет. В силу этого в окне **Harmonic (0=DC, 1=fundamental)**: блока **Fourier** укажем нечетные гармоники с 1 по 13 (Рис. 2.17).

В блоке **Scope** изменим название файла, в который записываются значения токов, на **ScopeData_2_5**.

Как и в случае с однополупериодным выпрямителем воспользуемся программой **Powergui** (Рис. 2.18). Полученные результаты свидетельствуют о том, что коэффициент нелинейных искажений в трехфазном мостовом выпрямителе меньше, чем в однополупериодном. Численные значения, полученные в ходе расчетов, представлены на рис. 2.19. Наибольшая гармоника – пятая. Её следует подавлять с помощью фильтров.

В библиотеке **SimPowerSystems Specialized Technology** фильтры реализованы в блоке **Three-Phase Harmonic Filter**. Демонстрационная модель **power_harmonicfilter**, которую можно запустить из окна команд, дает представление об использовании фильтров для подавления высших гармоник в энергосистеме.

DC component	= 1.316e-05	
Fundamental	= 218.7 peak (154.6 rms)	
THD	= 30.81%	
0 Hz (DC):	0.00%	270.0°
50 Hz (Fnd):	100.00%	-0.2°
100 Hz (h2):	0.01%	179.8°
150 Hz (h3):	0.11%	179.5°
200 Hz (h4):	0.01%	179.0°
250 Hz (h5):	22.70%	179.1°
300 Hz (h6):	0.01%	-1.2°
350 Hz (h7):	11.27%	178.8°
400 Hz (h8):	0.00%	177.5°
450 Hz (h9):	0.11%	-1.6°
500 Hz (h10):	0.00%	-2.6°
550 Hz (h11):	9.11%	-1.9°
600 Hz (h12):	0.01%	177.6°
650 Hz (h13):	6.42%	-2.2°
700 Hz (h14):	0.00%	-4.1°
750 Hz (h15):	0.11%	177.4°
800 Hz (h16):	0.00%	175.7°
850 Hz (h17):	5.71%	177.0°
900 Hz (h18):	0.00%	-3.6°
950 Hz (h19):	4.48%	176.8°

Рис. 2.19

2.2. Моделирование трансформаторов

В разделе **Elements** библиотеки **SimPowerSystems Specialized Technology** представлены многочисленные модели трансформаторов и автотрансформаторов, которые можно использовать при моделировании энергосистем и систем электроснабжения предприятий. Подробно рассмотрим некоторые из них.

2.2.1. Однофазный трансформатор

Начнем с блока силового трансформатора, в котором моделирование осуществляется без учета насыщения магнитной системы (**Linear Transformer**). Этот блок позволяет моделировать однофазные двух- или трехобмоточные трансформаторы.

Электромагнитная схема такого трансформатора представлена на рис. 2.20.

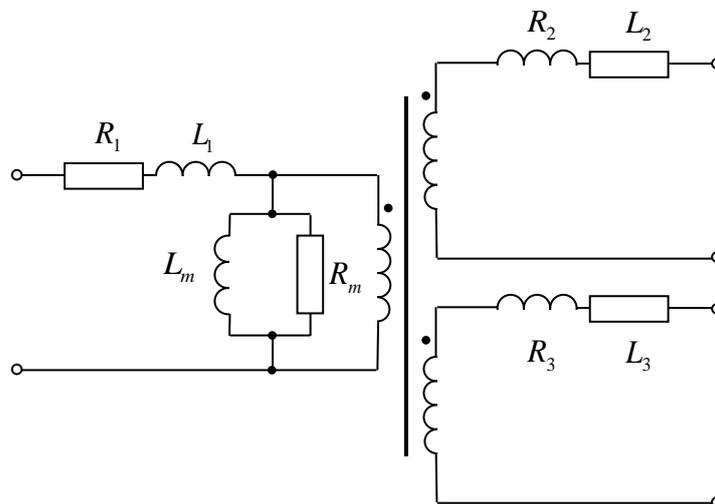


Рис. 2.20

Окно параметров блока с данными по умолчанию представлено на рис. 2.21.

Здесь:

- ✓ **Units** – единицы измерения параметров, которые могут задаваться в относительных единицах или в международной системе единиц (**SI**). По умолчанию используются относительные единицы (**pu**);
- ✓ **Nominal power and frequency [Pn(VA) fn(Hz)]**: – номинальные мощность (полная) и частота подводимого к трансформатору напряжения;
- ✓ **Winding 1 parameters [V1(Vrms) R1(pu) L1(pu)]**: – параметры первой обмотки: действующее значение напряжения (B), активное сопротивление (о.е.) и индуктивность рассеяния (о.е.) обмотки;

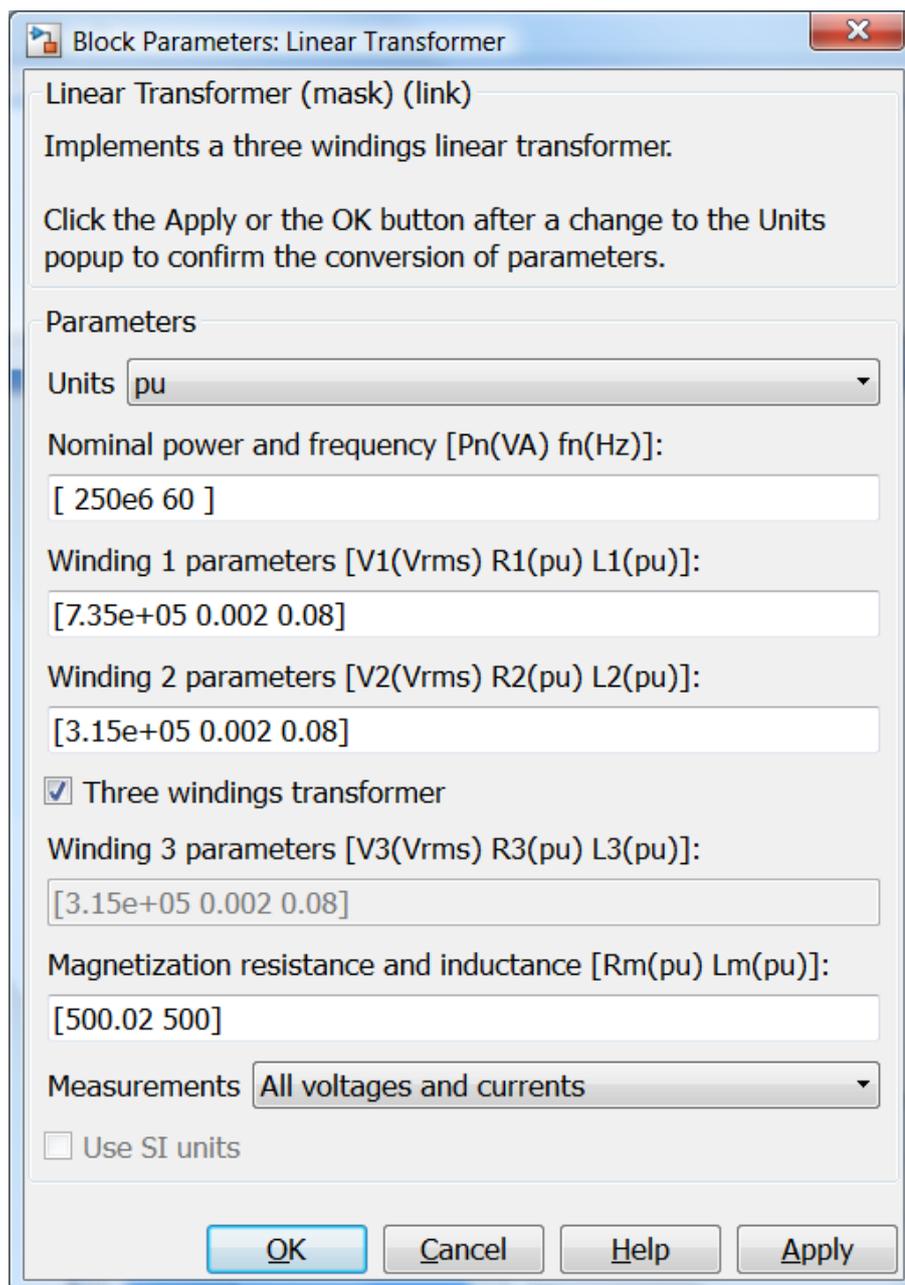


Рис. 2.21

- ✓ **Winding 2 parameters [V2(Vrms) R2(pu) L2(pu)]:** – параметры второй обмотки: действующее значение напряжения (B), активное сопротивление (о.е.) и индуктивность рассеяния (о.е.) обмотки;
- ✓ **Three windings transformer** – если флажок установлен, то трансформатор трехобмоточный, если нет – двухобмоточный;
- ✓ **Winding 3 parameters [V3(Vrms) R3(pu) L3(pu)]:** – параметры третьей обмотки: действующее значение напряжения (B), активное сопротивление (о.е.) и индуктивность рассеяния (о.е.) обмотки;

- ✓ **Magnetization resistance and inductance [Rm(pu) Lm(pu)]:** – активное сопротивление и взаимная индуктивность обмоток намагничивающей ветви (рис. 2.20);
- ✓ **Measurements** – измеряемые параметры выбираются из выпадающего списка:
 - **Winding voltages** – напряжения обмоток;
 - **Winding currents** – токи обмоток;
 - **Magnetization current** – ток намагничивания;
 - **All voltages and currents** – все напряжения и токи.

Пример 2.6.

Рассчитать ток короткого замыкания в первой и во второй обмотках трансформатора ТМ-1000/10 из *примера 1.3*, воспользовавшись блоком **Linear Transformer**.

Решение

Определяем параметры обмоток трансформатора и намагничивающей ветви с учетом следующих допущений: параметры первой обмотки и приведенные параметры второй обмотки равны; при подсчете номинального тока пренебрегаем намагничивающим током; на холостом ходу ЭДС и напряжение первой обмотки равны друг другу.

С учетом принятых допущений и на основании данных, представленных в таблице 1.1 (стр. 20), находим активные сопротивления и индуктивности рассеяния обмоток трансформатора, а также параметры намагничивающей ветви:

$$I_{1н} \approx \frac{S_{2н}}{\sqrt{3}U_{1н}} = \frac{1000}{\sqrt{3} \cdot 10} = 57,7367 \text{ А} \text{ – номинальный фазный ток первичной}$$

обмотки;

$$z_{\kappa} = \frac{U_{1н} \cdot u_{\kappa} \%}{\sqrt{3}I_{1н} 100\%} = \frac{10000 \cdot 5,5\%}{\sqrt{3} \cdot 57,7367 \cdot 100\%} = 5,50 \text{ Ом} \text{ – полное сопротивление}$$

короткого замыкания;

$$r_{\kappa} = \frac{P_{\kappa}}{3I_{1н}^2} = \frac{12000}{3 \cdot 57,7367^2} = 1,1999 \text{ Ом} \text{ – активное сопротивление короткого}$$

замыкания;

$$R_1 \approx R_2' = \frac{r_{\kappa}}{2} = \frac{1,1999}{2} = 0,59995 \text{ Ом} \text{ – активное сопротивление первой}$$

обмотки и приведенное активное сопротивление второй;

$$R_2 = \frac{R_2'}{k^2} = \frac{0,59995}{25^2} = 0,0009599 \text{ Ом} \text{ – активное сопротивление второй}$$

обмотки, здесь $k \approx 10/0,4 = 25$ – коэффициент трансформации;

$x_k = \sqrt{z_k^2 - r_k^2} = \sqrt{5,5^2 - 1,1999^2} = 5,3675 \text{ Ом}$ – реактивное сопротивление короткого замыкания;

$L_1 \approx L_2' = \frac{x_k}{2\omega} = \frac{5,3675}{2 \cdot 2\pi \cdot 50} = 8,5426 \text{ мГн}$ – индуктивность рассеяния первой обмотки и приведенная индуктивность рассеяния второй обмотки;

$L_2 = \frac{L_2'}{k^2} = \frac{8,5426}{25^2} = 0,01366848 \text{ мГн}$ – индуктивность рассеяния второй обмотки;

$P_{xx} = 3E_{\phi 1} I_{xxa} \approx 3U_{\phi 1} \frac{U_{\phi 1}}{R_m} \Rightarrow R_m = \frac{U_1^2}{P_{xx}} = \frac{10000^2}{1900} = 52631,58 \text{ Ом}$ – активное

сопротивление параллельной ветви намагничивающего контура схемы замещения, потери в которой равны потерям холостого хода (магнитным потерям);

$I_{xxa} = \frac{U_{\phi 1}}{R_m} = \frac{U_1}{\sqrt{3} \cdot R_m} = \frac{10000}{\sqrt{3} \cdot 52631,58} = 0,1097 \text{ А}$ – активная составляющая

тока холостого хода;

$I_{xx} = I_{1n} \frac{I_{xx} \%}{100\%} = 57,7367 \frac{1,7}{100} = 0,9815 \text{ А}$ – ток холостого хода;

$I_{xxr} = \sqrt{I_{xx}^2 - I_{xxa}^2} = \sqrt{0,9815^2 - 0,1097^2} = 0,9754 \text{ А}$ – реактивная (намагничивающая) составляющая тока холостого хода;

$L_m = \frac{U_{\phi 1}}{\omega I_{xxr}} = \frac{U_1}{\sqrt{3}\omega I_{xxr}} = \frac{10000}{\sqrt{3} \cdot 314,159 \cdot 0,9754} = 18,84169 \text{ Гн}$ – взаимная

индуктивность обмоток.

Создаем ново окно **SimPowerSystems** модели: Переносим в него из библиотеки **Simulink** по одному экземпляру следующие блоки: (подраздел *Commonly Used Blocks*) **Scope**; **Mux**; **Gain**; (*Sinks*) **Display**; (*Math Operations*) **Real-Imag to Complex**; **Abs**; (*Sources*) **Step**. Из библиотеки **SimPowerSystems** → **Specialized Technology** перенесем блоки: (*Electrical Sources*) **AC Voltage source**; (*Measurements*) **Voltage Measurement**; **Current Measurement**; (*Control and Measurements Library* → *Measurements*) **Power**; **RMS**; (*Elements*) **Breaker**, **Linear Transformer**, **Three-Phase Series RLC Branch**.

После того, как блоки перемещены в окно модели в блоках **Power** и **RMS**, необходимо поменять частоту 60 Гц на 50 Гц , в блоке **Scope** сделать два входа (*две оси*), в блок **Gain** внести цифру **3** для расчета мощности трех фаз. После этого блоки можно копировать. Для этого необходимо, удерживая на клавиатуре клавишу **Ctrl**, выделить курсором блок.

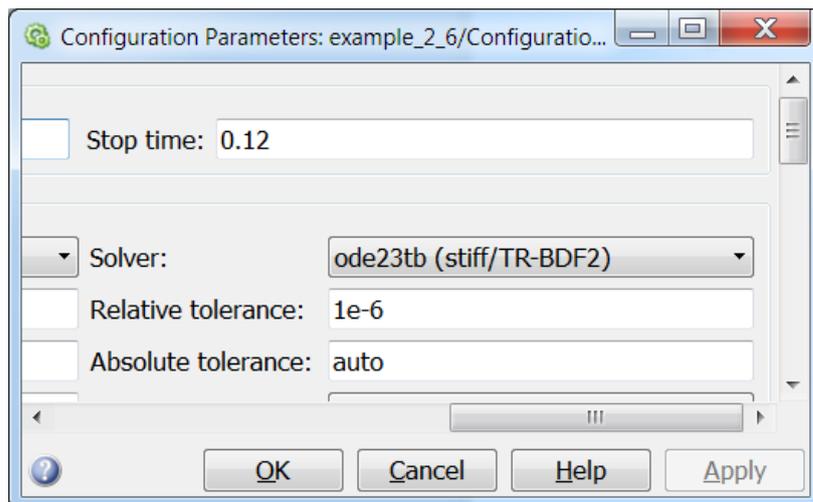


Рис. 2.22

Нажать левую клавишу мышки, перетащить копию блока в требуемую позицию, отпустить клавишу **Ctrl** и левую клавишу мышки.

Далее собираем схему, представленную на рис. 2.23, и сохраняем её в файле **example_2_6**. Время расчета задаем равное **0.12 с**. Во вкладке **Simulation** → **Model Configuration Parameters** при необходимости вносим изменения по решателю и относительной точности согласно рис. 2.22. В блоке **Step** устанавливаем **Step time** равное 0.02 с.

Поскольку рассматривается одна фаза трехфазного трансформатора, то в окно параметров источника переменного напряжения необходимо ввести амплитудное значение фазного напряжения $U_{m\phi 1} = \sqrt{2}U_1/\sqrt{3}$, т.е. $\text{sqrt}(2)*10000/\text{sqrt}(3)$, и частоту 50 Гц.

В окне параметров блока сопротивлений выберем тип ветви **R** и внесем в окно **Resistance (Ohms)** три буквы **inf**, что означает бесконечность. Последнее необходимо для моделирования установившегося режима холостого хода. Запускаем модель. Результаты этого моделирования представлены на дисплеях рис. 2.23. В верхнем правом углу на **Display P_1, Q_1, S_1** показаны значения мощностей. Как видим, эти потери не отличаются от паспортных значений, а токи холостого хода достаточно близки (Смотри **Display I_1** в верхнем левом углу). Следует отметить значительную реактивную мощность, потребляемую трансформатором в режиме холостого хода (16,9 кВАр). В блоке **Display P_1, Q_1, S_1** установлен формат **long**.

Проведем опыт короткого замыкания. Для этого изменим сопротивление нагрузки на **1e-12 Ом**, **Step time** на **0.02 с**, напряжение источника уменьшим до напряжения короткого замыкания, т.е. до $\text{sqrt}(2)*10000/\text{sqrt}(3)*0.055$ В, а относительную точность установим **1e-8**.

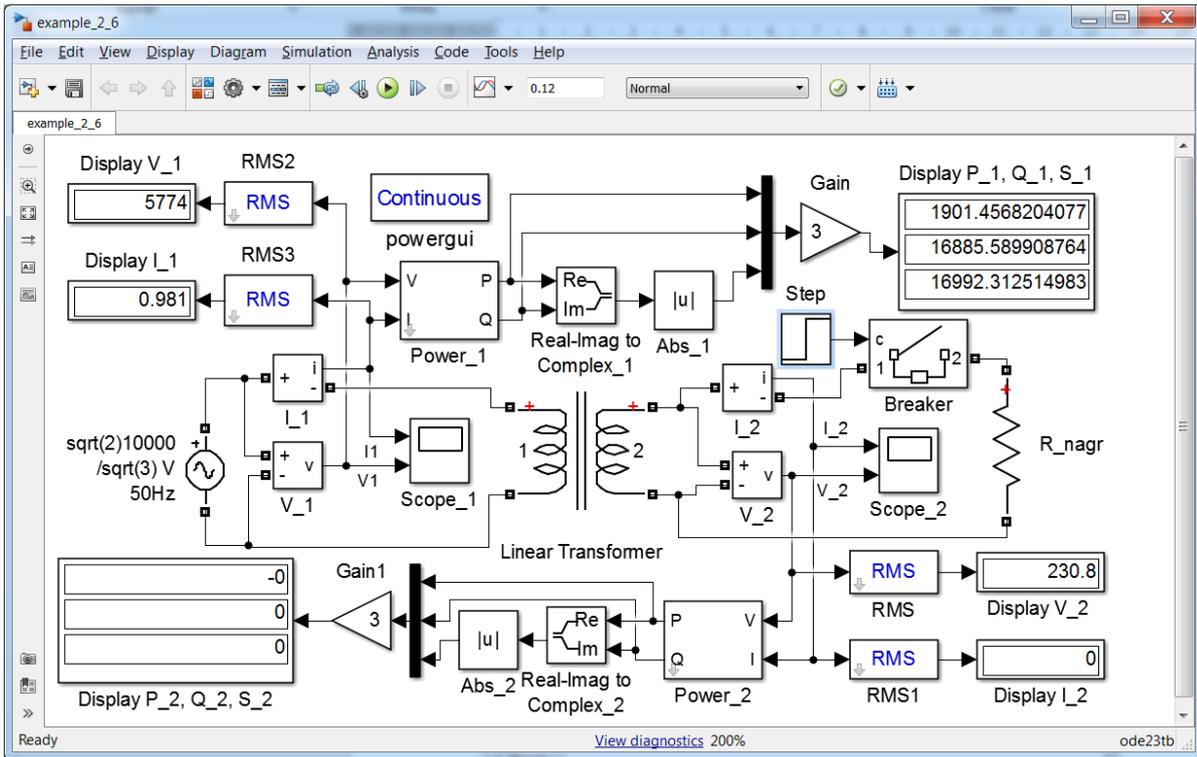


Рис. 2.23

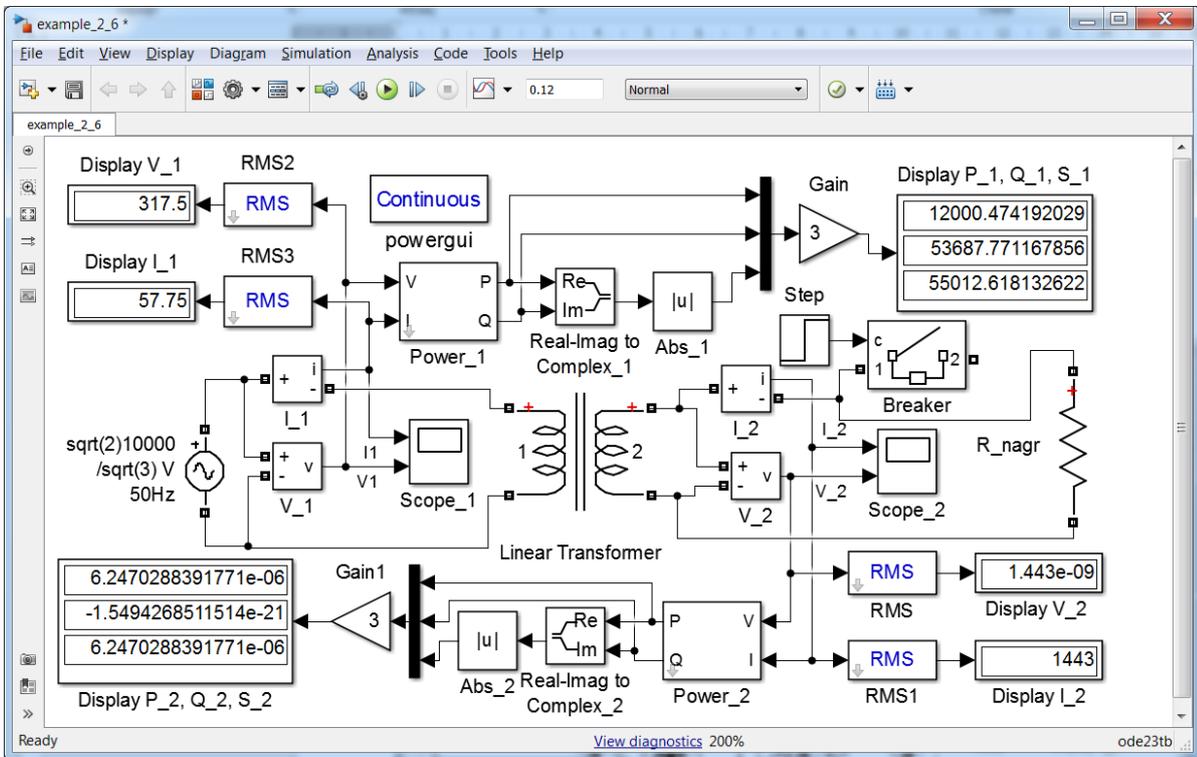


Рис. 2.24

Результаты расчета, представленные в дисплеях на рис. 2.24, показывают, что потери короткого замыкания совпали с паспортными данными, а токи обмоток также достаточно близки паспортным значениям, что свидетельствует о хорошей сходимости.

Следует отметить, что в модели имеется возможность рассчитать токи и напряжения в установившийся режим работы, не прибегая к решению дифференциальной системы уравнений, а используя алгебраические уравнения. Для этого двойным щелчком мышки по блоку **Continuous** вызываем программу **powergui**, нажимаем на кнопку **Steady-State Voltages and Currents**. При необходимости изменяем параметры схемы, например, замыкаем вторичную обмотку, нажимаем **Update Steady State values** и получаем результаты, представленные на рис. 2.25.

Расчет переходного процесса короткого замыкания осуществим при неблагоприятной фазе $\alpha_0 = \varphi_k + \pi/2 = 1.3509 + \pi/2 = 2.9217$. Введем это значение фазового сдвига в окно параметров источника переменного напряжения, преобразовав радианы в градусы, т.е. $2.9217 \cdot 180/\pi$. На рис. 2.26 представлены кривые переходного процесса во второй обмотке трансформатора. Значения токов и напряжений совпали с результатами предыдущих расчетов, полученными с помощью **MATLAB** и **Simulink** (Рис. 1.16 и Рис. 1.29). Кривая напряжения в данном случае представлена более корректно, т.к. при коротком замыкании напряжение равно нулю, при условии, что равно нулю переходное сопротивление в месте КЗ.

Следует отметить, что и кривая тока в первой обмотке качественно имеет такой же вид, как и во второй обмотке (Рис. 2.27).

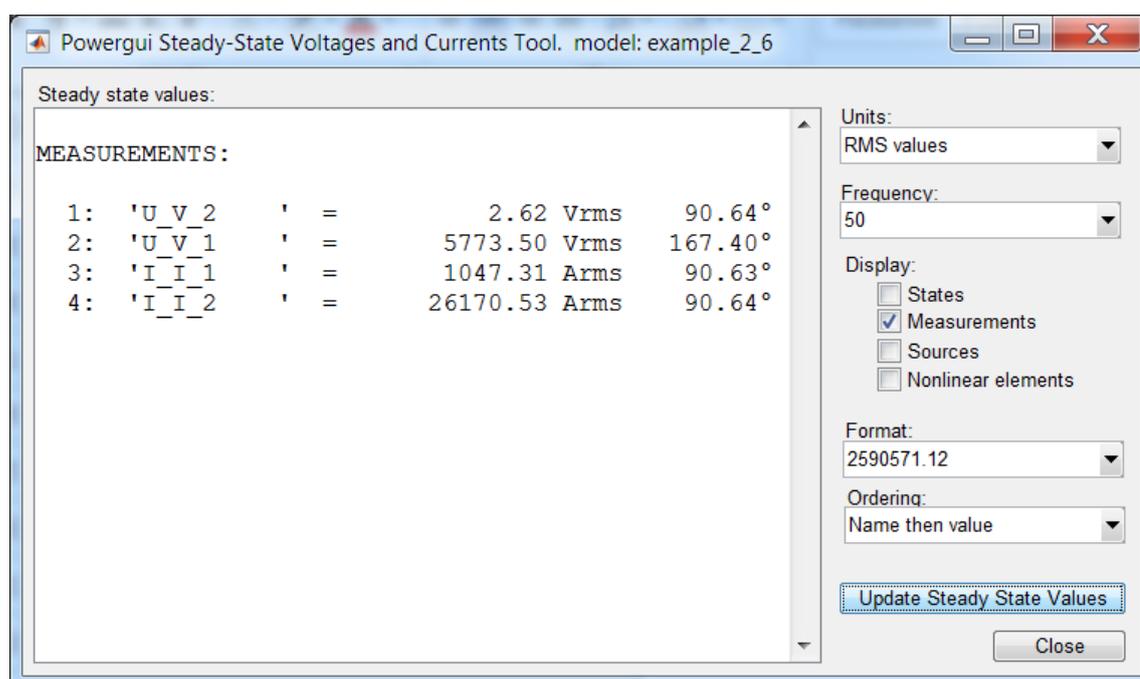


Рис. 2.25

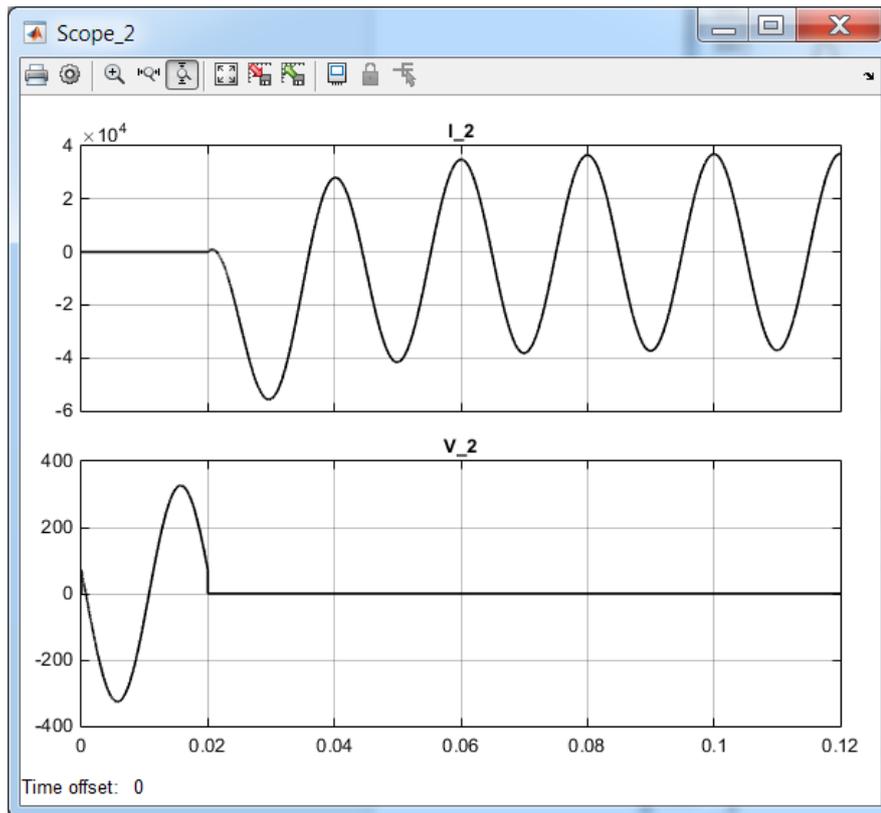


Рис. 2.26

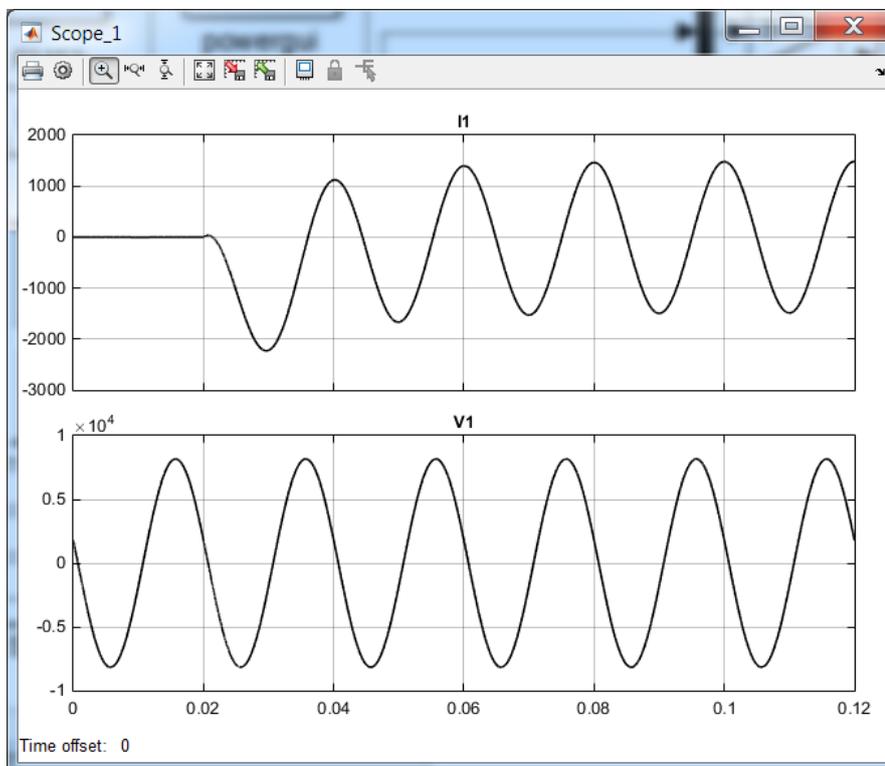


Рис. 2.27

Примечание. При разработке данной модели возникли проблемы. При непосредственном подключении ко второй обмотке трансформатора активного сопротивления $R_{nagr} = 1e-12$ (режим короткого замыкания) вопреки ожиданиям в кривой тока отсутствовала апериодическая составляющая. С другой стороны, при $R_{nagr} = 1e+12$ (режим холостого хода) появлялась неизменная апериодическая составляющая. Обойти эти проблемы удалось с помощью управляемого блока **Breaker**, на который подавался сигнал от **Step** в нулевой (или чуть больше) момент времени.

Также решить эти проблемы удалось с помощью программы **powergui**, во вкладке которой имеется кнопка **Initial States Setting** (Задание начальных значений), открывающая окно, в котором нужно отметить **To Zero** (нулевые начальные значения) для режима короткого замыкания или **To Steady State** (Начальные значения установившегося режима) для режима холостого хода. Такая же проблема, но для цепи постоянного тока, была описана в [2]. Естественно, что первый способ решения этих проблем более предпочтителен.

2.2.2. Трехфазный трансформатор

Существенно большие возможности по расчету систем электроснабжения предоставляют многочисленные блоки трехфазных трансформаторов (автотрансформаторов). Рассмотрим подробно один из них, например **Three-Phase Transformer (Two Windings)** – трехфазный двухобмоточный трансформатор с двумя обмотками. В модели учитывается нелинейность характеристики намагничивания магнитопровода. В первой вкладке окна параметров блока (рис. 2.28) для каждой обмотки выбирают из выпадающего списка схему соединения фаз обмотки (звезда или треугольник) и группу соединения:

- Y – соединение фаз обмотки в звезду;
- Yn – звезда с доступом к нейтрали;
- Yg – звезда с заземленной нейтралью;
- Delta (D1) – соединение фаз обмотки в треугольник, группа соединения – "первая";
- Delta (D11) – соединение фаз обмотки в треугольник, группа соединения – "одиннадцатая".

Следует отметить, что стандартными являются следующие группы соединения: Y/Y-0 (нулевая); D/Y-11; Y/D-11. Расчет с учетом нелинейной характеристики намагничивания осуществляется, если выбрана позиция **Saturable core**. После её активизации открываются ещё две позиции **Simulate hysteresis** (моделирование гистерезиса) и **Specify initial fluxes** (задать начальные потоки).

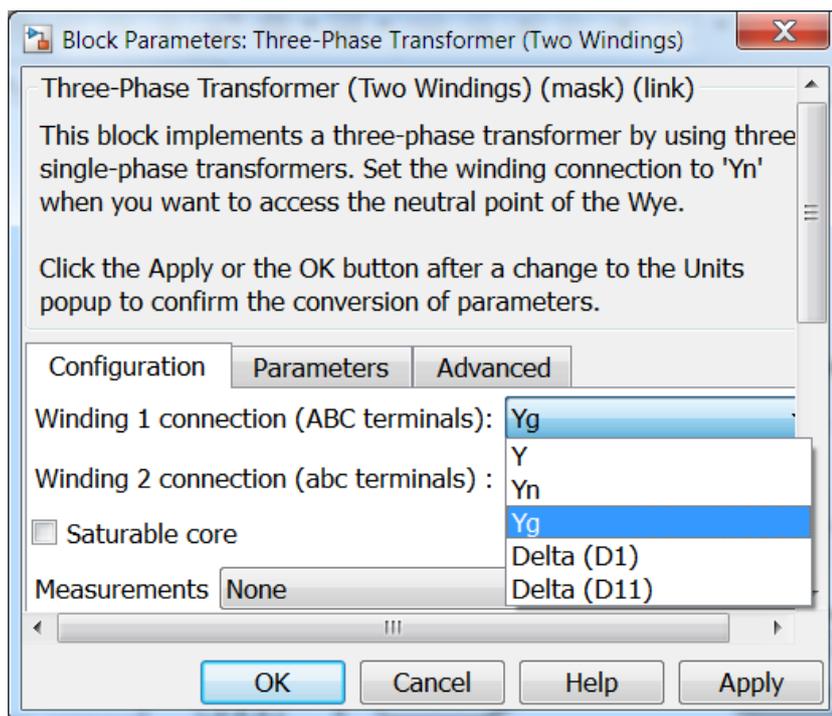


Рис. 2.28

При выборе первой из них необходимо использовать файл с данными кривой гистерезиса при работе с **powergui**, а при выборе второй – откроется доступ к окну во вкладке **Parameters**, в которой можно задать начальные значения потока по фазам (остаточный поток). В выпадающем окне позиции **Measurements** можно выбрать переменные, которые будут доступны для отображения.

Во вкладке **Parameters** (Рис. 2.29) выбираем систему SI (СИ) или pu (относительные единицы), в которой будут представлены параметры трансформатора. Далее следуют окна, в которые следует занести соответствующие параметры:

- **Nominal power and frequency [Pn(VA) , fn(Hz)]** – номинальная мощность и частота;
- **Winding 1 parameters [V1 Ph-Ph(Vrms) , R1(Ohm) , L1(H)]** – параметры первой обмотки: линейное напряжение (действующее); активное сопротивление обмотки (Ом); индуктивность рассеяния (Гн);
- **Winding 2 parameters [V2 Ph-Ph(Vrms) , R2(Ohm) , L2(H)]** – аналогичные данные второй обмотки;
- **Magnetization resistance Rm (Ohm)** – активное сопротивление ветви намагничивания (Ом);
- **Magnetization inductance Lm (H)** – индуктивность (взаимная индуктивность) намагничивающего контура;

- **Saturation characteristic** – характеристика насыщения (кривая намагничивания);
- **Initial fluxes** – начальные значения потоков.

Во вкладке **Advanced** (продвинутый) можно установить дополнительный параметр – разрыв алгебраического контура в дискретной модели.

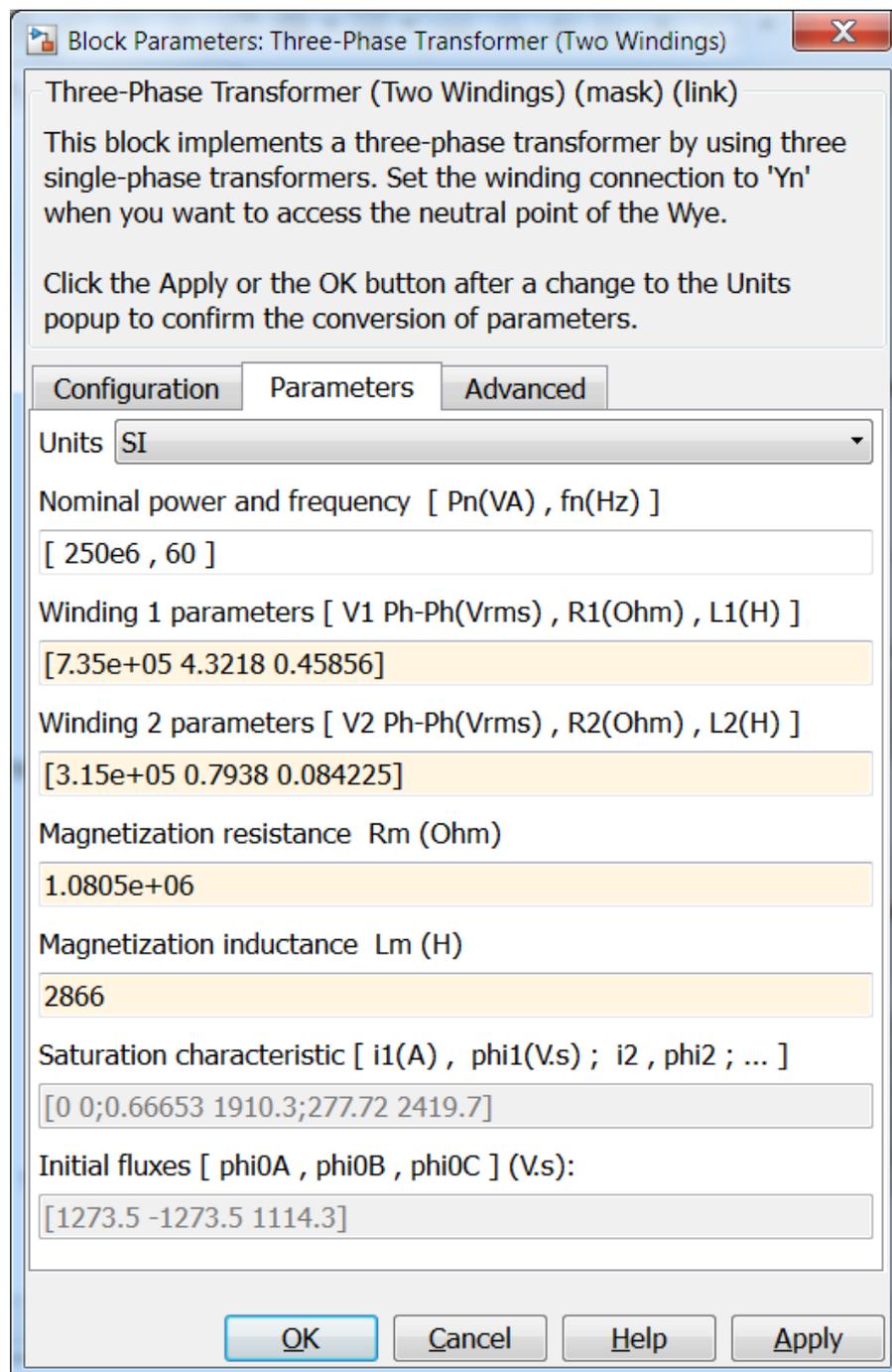


Рис. 2.29

Рассмотрим пример модели с этим блоком для расчета токов короткого замыкания.

Пример 2.7.

Рассчитать динамические режимы при трех-, двух - и однофазных коротких замыканиях трехфазного двухобмоточного трансформатора с параметрами из примера *Пример 2.6.*

Решение

Создаем новую модель, которую сохраняем в файле **example_2_7**.

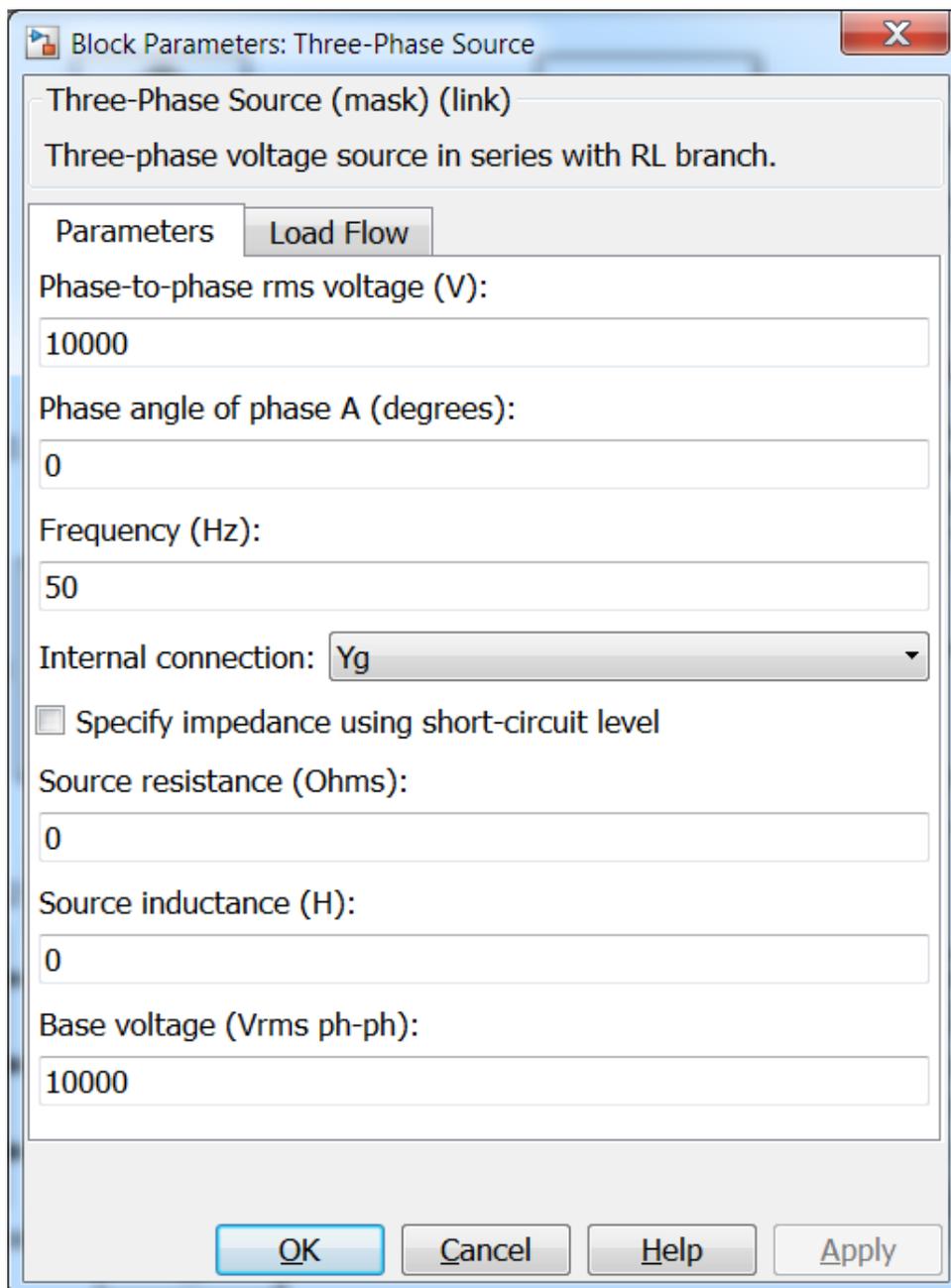


Рис. 2.30

Переносим в неё из библиотеки **Simulink** по одному экземпляру следующих блоков: (подраздел *Commonly Used Blocks*) **Scope**; **Mux**, (*Signal Routing*) **From**, (*Sinks*) **Display**; (*Math Operations*) **Real-Imag to Complex**; **Abs**. Из библиотеки **SimPowerSystems** → **Specialized Technology** перенесем блоки: (*Electrical Sources*) **Three-Phase Source**; (*Measurements*) **Current Measurement**, **Three-Phase V-I Measurement**, **Voltage Measurement**; (*Control and Measurements Library* → *Measurements*) **Power (3ph, Instantaneous)**, **RMS**; (*Elements*) **Three-Phase Fault**, **Three-Phase Series RLC Load**, **Three-Phase Transformer (Two Windings)**.

В блоке **Three-Phase Source** задаем параметры согласно рис. 2.30. В окне **Phase angle of phase A (degrees)**: занесено численное значение неблагоприятной фазы короткого замыкания. Поскольку в предыдущих примерах расчета использовался идеальный источник напряжения, то в этом блоке сопротивления источника задаем равные нулю.

В блоке измерения трехфазных напряжений и токов **Three-Phase V-I Measurement** вносим изменения согласно рис. 2.31.

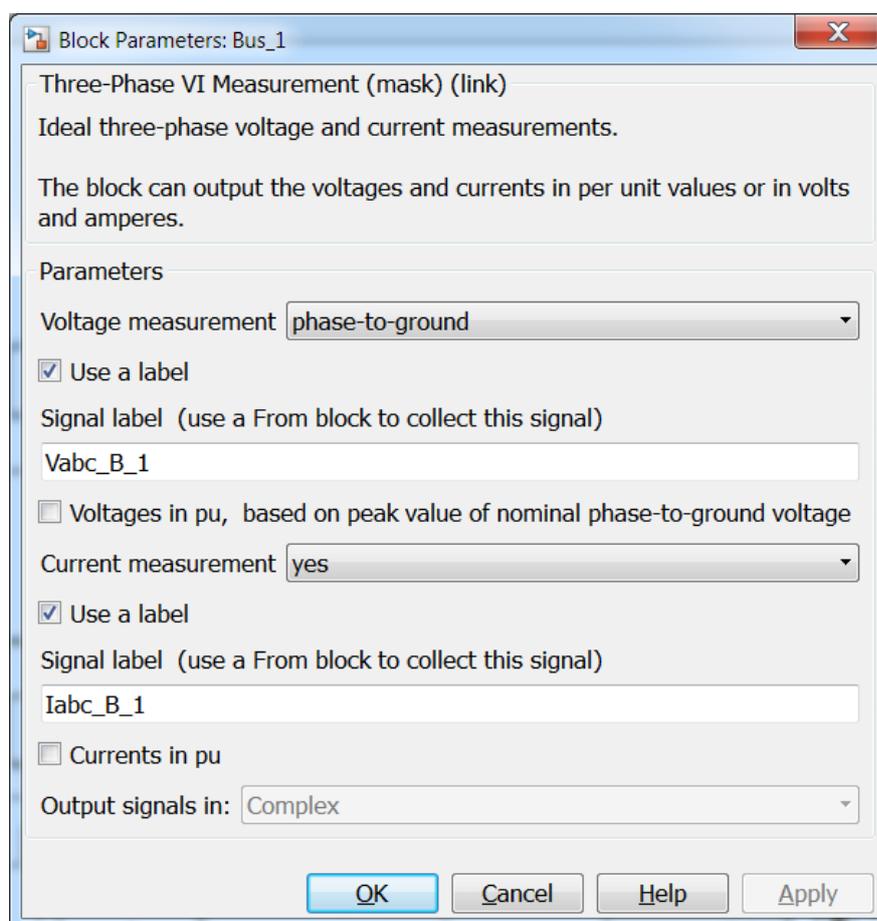


Рис. 2.31

Копируем его и изменяем название блока на **Bus_1** (шина первой обмотки, шина высокого напряжения). После этого копируем этот блок. Копию блока переименовываем на **Bus_2** (шина низкого напряжения). Открываем окно его параметров, в окне **Signal label** (обозначение сигнала) заменяем цифру 1 на 2 и получаем **Vabc_B_2**. Аналогично поступаем с обозначением тока.

В окне параметров блока **Three-Phase Transformer (Two Windings)** по вкладке **Configuration** устанавливаем соединение **Y/Yn-0**. Во вкладке **Parameters** вносим численные значения параметров обмоток и намагничивающей ветви согласно рис. 2.32. Изменяем название блока трансформатора на **TM-1000/10**.

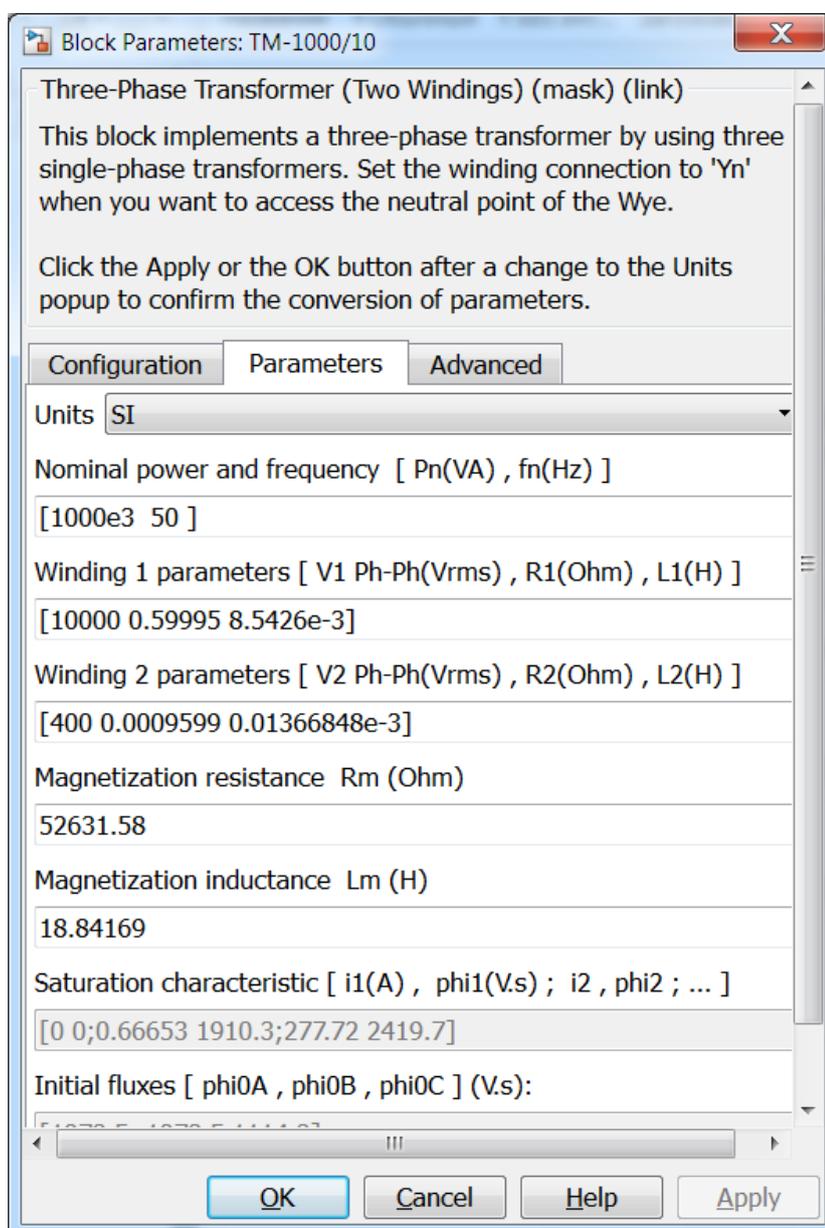


Рис. 2.32

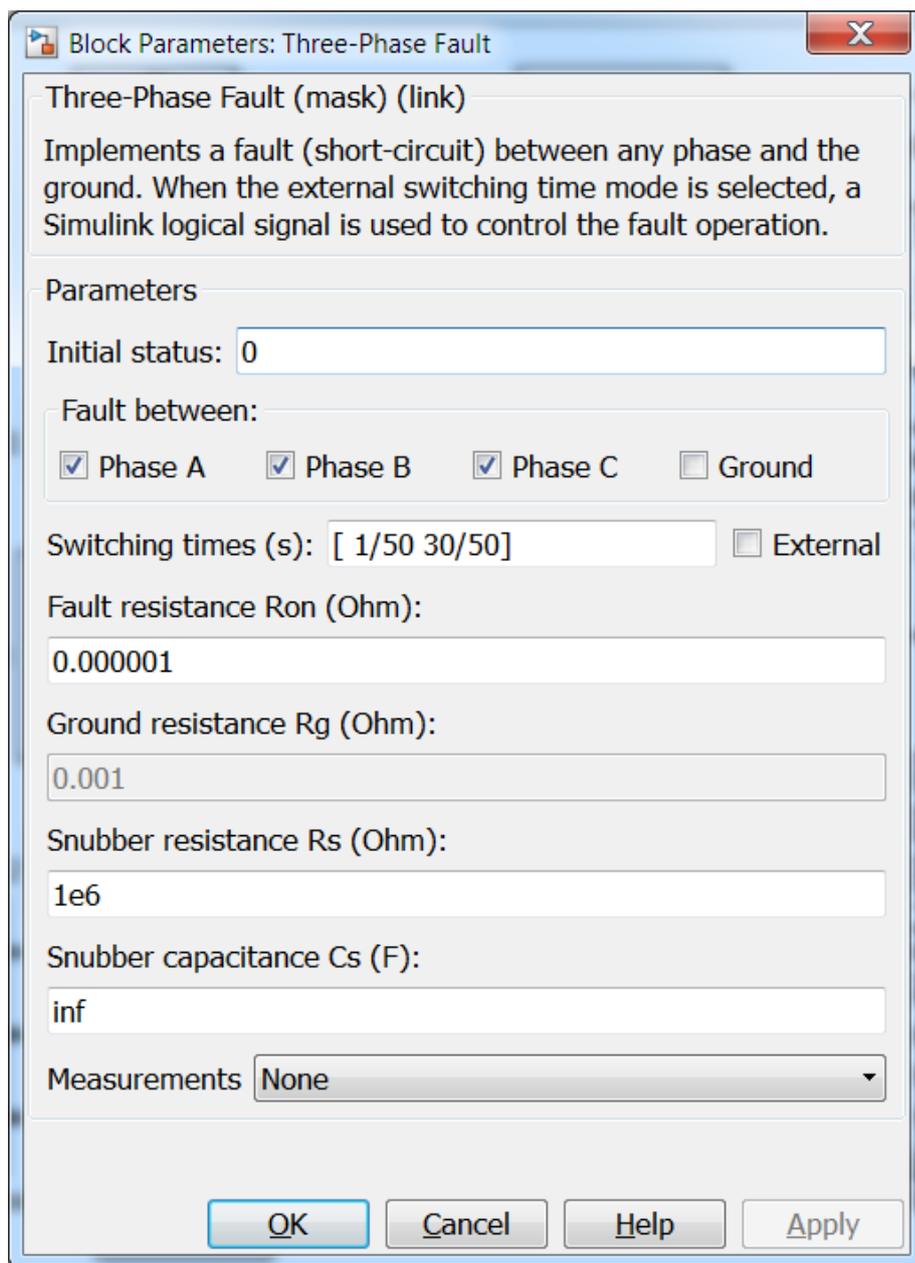


Рис. 2.33

В окне параметров блока **Three-Phase Fault** вносим изменения согласно рис. 2.33. Поскольку моделируем трехфазное короткое замыкание, то флажками необходимо отметить три фазы, если флажками отметить фазу **A** и землю (**Ground**), то можно рассчитать однофазное короткое замыкание, и т.д. Таким образом, этот блок позволяет создавать различные короткие замыкания в моделях. Время замыкания и размыкания задается в данном случае в виде периодов.

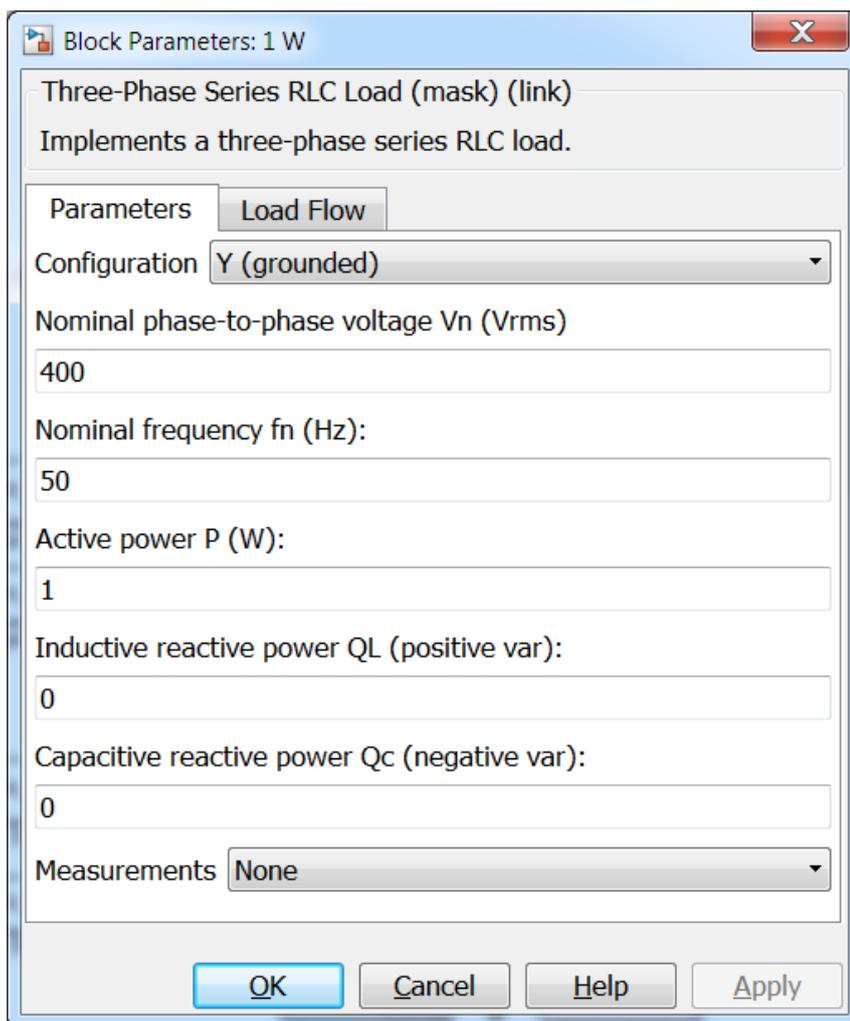


Рис. 2.34

Имеется возможность управления этим процессом в помощь внешнего источника сигнала (**External**). Значения демпфирующих резистора и конденсатора по умолчанию можно оставить.

Блок **Three-Phase Series RLC Load** носит вспомогательный характер (Рис. 2.34). При его отсутствии модель работает некорректно, а уменьшение мощности менее 1 W приводит к значительному росту времени расчета.

Для получения сигналов от блоков **Bus_1** и **Bus_2** открываем окно параметров блока **From** (*принимать от*), в позиции **Goto tag:** (передаваемый тег, метка) заносим **Vabc_B_1**. Изменяем его название на **From Bus_1_V**. Копируем этот блок. Изменяем в окне параметров **Vabc_B_1** на **Iabc_B_1**. Его название изменяем на **From Bus_1_I**. После этого каждый из блоков копируем и заменяем в окне параметров и в их названии цифру **1** на **2**.

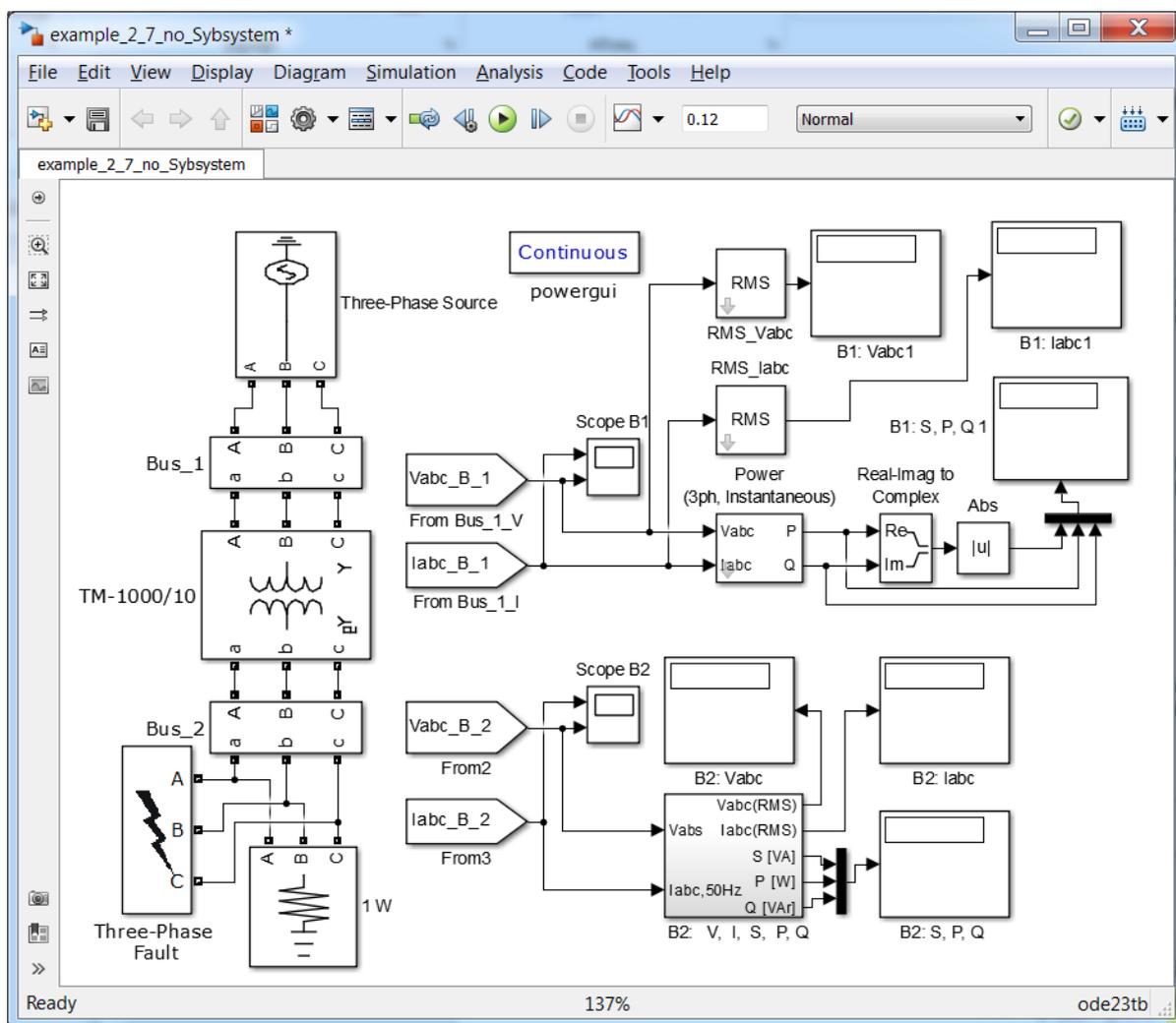


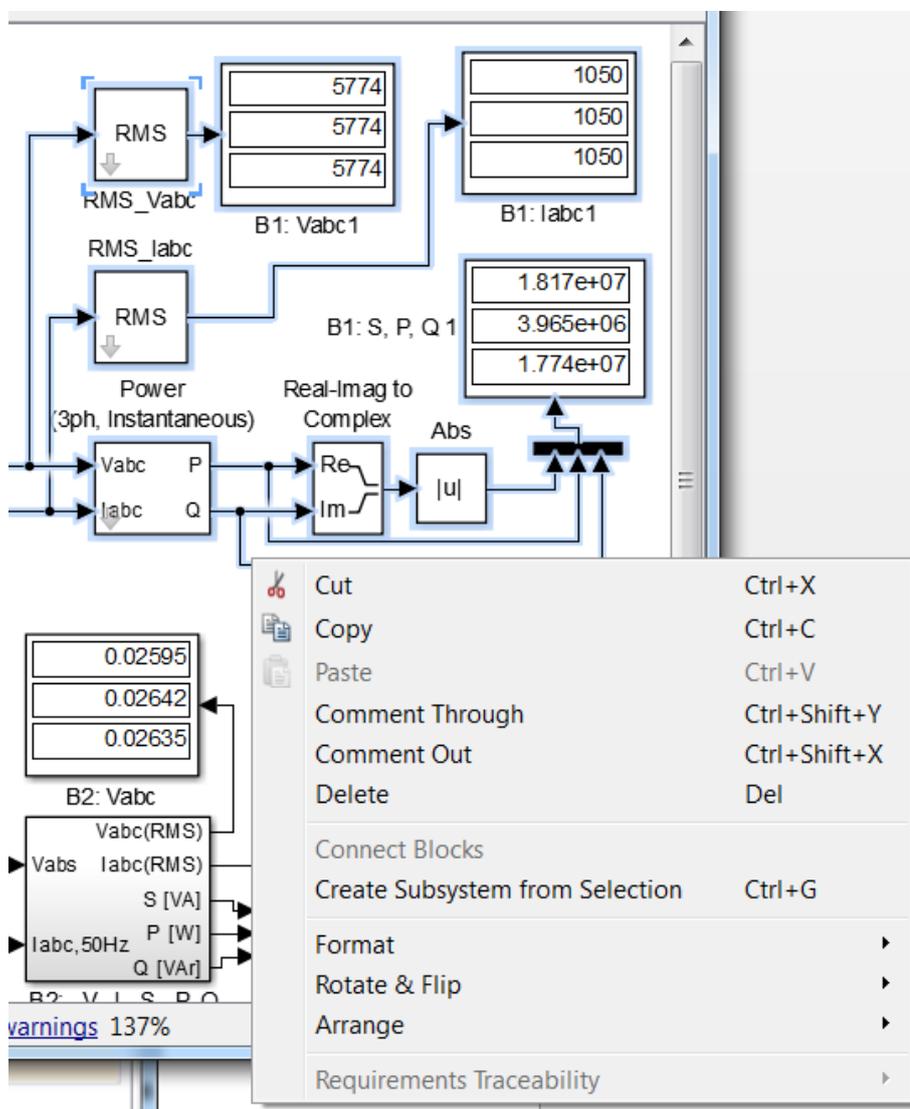
Рис. 2.35

Полученные четыре блока организуют "беспроводную" связь с соответствующими блоками измерения трехфазных напряжений и токов **Bus_1** и **Bus_2** (Рис. 2.35). В дальнейшем эти данные необходимо визуализировать. Для этого воспользуемся блоками **Scope B1** и **Scope B2**, на которые выведем кривые тока и напряжений с соответствующих шин **Bus_1** и **Bus_2**. Следует отметить, что достоинства "беспроводной" связи возрастают по мере усложнения модели и роста числа измеряемых величин.

Каждый переходной режим обычно завершается установившимся режимом, наступление которого легко отследить по осциллограммам. В этом режиме интерес представляют действующие значения синусоидальных напряжений и токов, а также соответствующие мощности на шинах трансформатора. Для получения действующих значений синусоидальных напряжений и токов воспользуемся блоком

RMS, в окно параметров которого введем частоту **50 Гц**. Скопируем этот блок. Изменим названия блоков на **RMS_Vabc** и **RMS_Iabc**. Блоки дисплеев обозначим как **B1: Vabc1**, **B1: Iabc1** и **B1: S, P, Q 1**. После этого последовательно соединим блоки **Vabc_B_1**, **RMS_Vabc** и **B1: Vabc1**, связанные с измерением напряжения. Аналогичным образом соединим блоки, связанные с выводом трехфазных токов (Рис. 2.35).

Для измерения активной и реактивной мощностей воспользуемся блоком **Power (3ph, Instantaneous)**. Используя эти мощности, определим полную мощность как модуль комплексного числа: $|S| = \text{abs}(P + jQ)$ с помощью блоков **Real-Imag to Complex** и **Abs**. К дисплею **B1: S, P, Q 1** подводим значения мощностей. Следует отметить, что такой подход уже использовался в *примере 2.6*, но только для однофазной цепи.



ис. 2.36

По мере усложнения основной модели возникает необходимость в объединении простых блоков модели в подсистему (**Subsystem**), что позволяет разбить основную модель на несколько подсистем, которая в свою очередь может включать свои подсистемы и т.д.

Такая подсистема **B2: V, I, S, P, Q** используется для измерения напряжений, токов и мощностей на второй шине. К этой подсистеме подаются мгновенные значения трехфазных напряжений и токов, а на выходе получаем их действующие значения и мощности. Таким образом, эта подсистема выполняет функции пяти блоков, которые используются при измерении на первой шине.

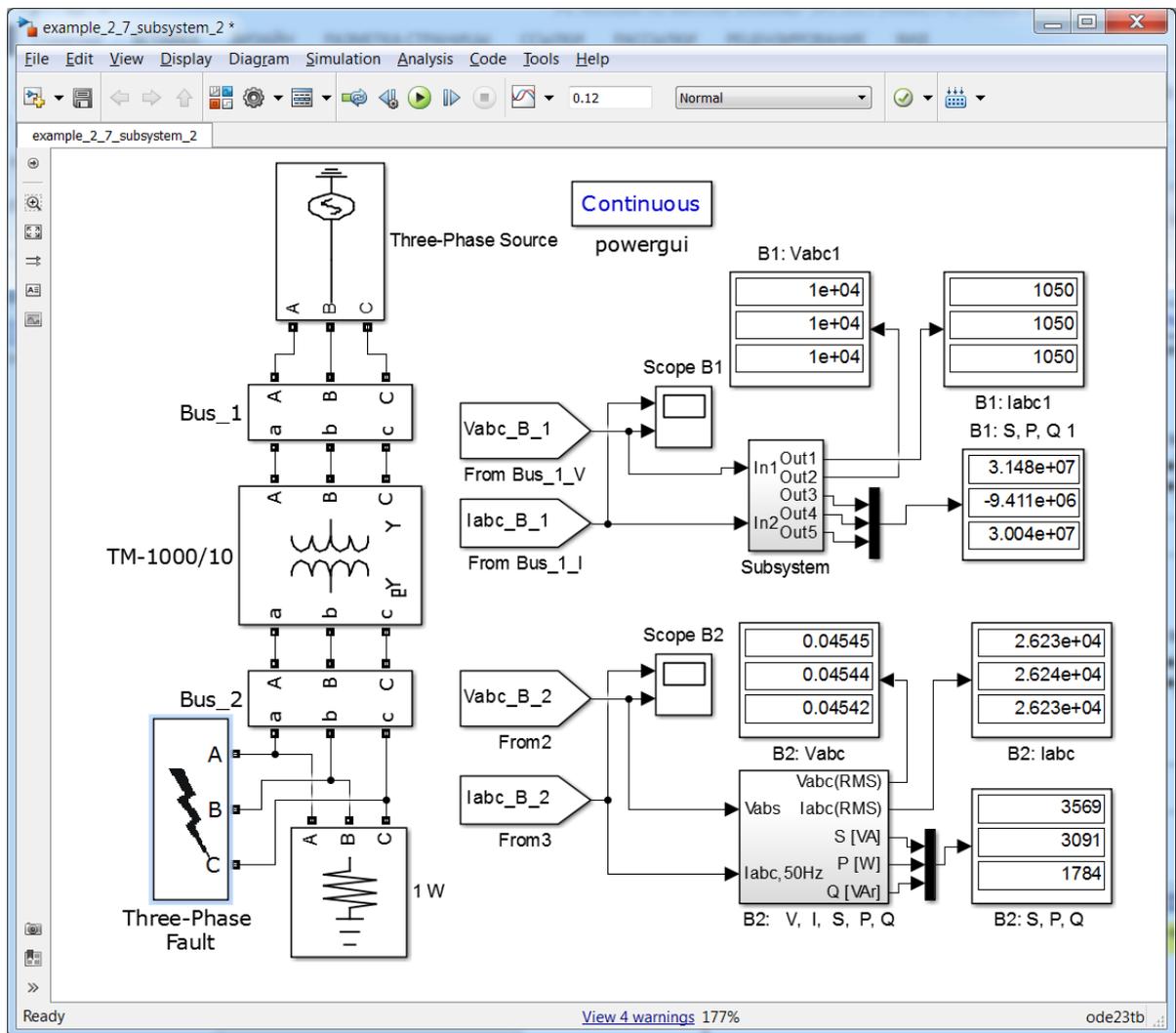


Рис. 2.37

Для создания подсистемы достаточно выполнить следующие простые действия:

- Выделить линии входа, выхода и блоки основной модели, которые необходимо поместить в подсистему. Для этого нажимаем левую кнопку мыши и перемещаем её, выделяя необходимые блоки и линии. Для "прицельного" выделения удерживаем клавишу **Shift**, поочередно наводим курсор на требуемый блок или линию и щелкаем левой кнопкой мыши;
- После этого нажимаем правую кнопку мыши. В выпадающем списке переходим на **Create Subsystem from Selection** (создать подсистему из выбранного) и создаем подсистему (рис. 2.36). При желании для этой цели можно воспользоваться и клавиатурой, нажав на **Ctrl+G**.

После незначительной редакторской правки модель с подсистемами принимает более изящный вид (рис. 2.37). В принципе можно оставить только осциллографы и дисплеи, а остальное "спрятать" в подсистему.

Следует отметить, что одним из достоинств трехфазной модели, по сравнению с однофазной, является то, что с её помощью можно рассчитывать как двухфазные короткие замыкания (рис. 2.38), так и однофазные короткие замыкания, в частности, при соединении D/Yn-11 (рис. 2.39).

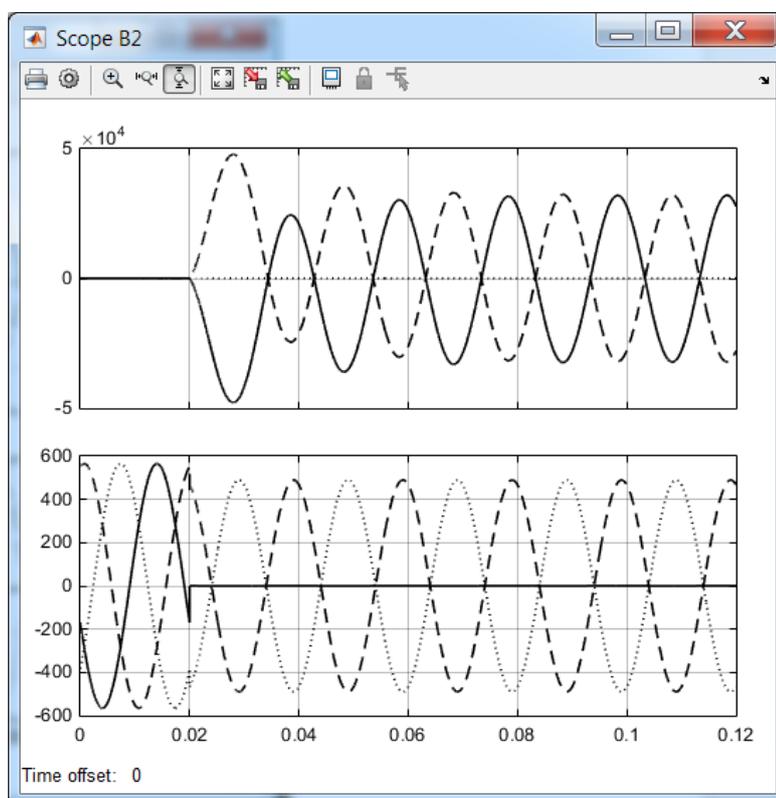


Рис. 2.38

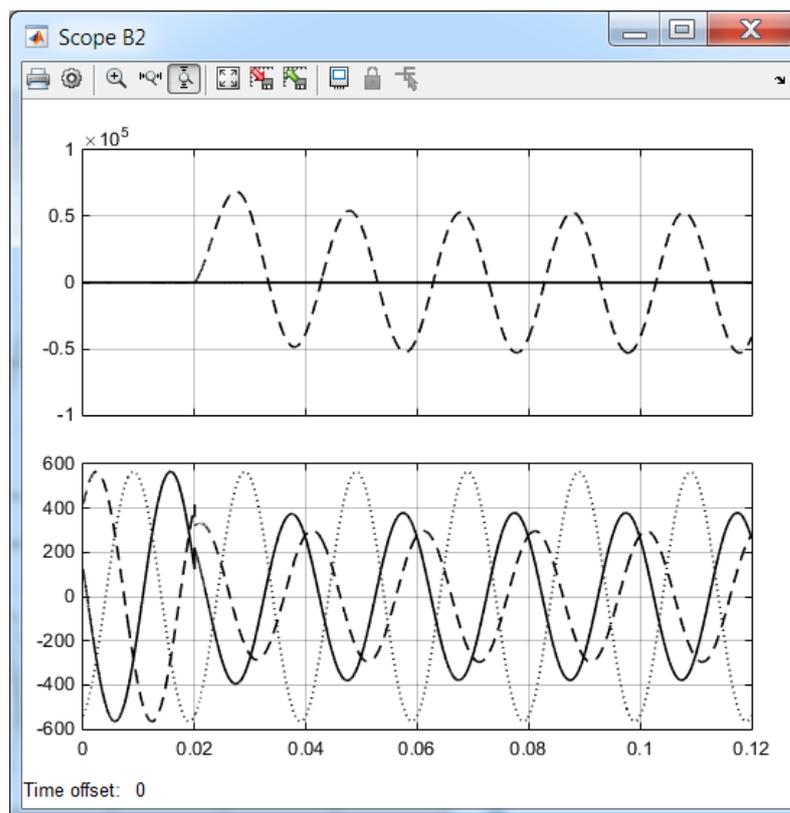


Рис. 2.39

Модель позволяет исследовать влияние нагрузки на цеховой трансформатор, в том числе подключение компенсирующих устройств и асинхронных двигателей.

Последние являются основными потребителями электроэнергии в цеховых сетях промышленных предприятий. В связи с этим рассмотрим один из блоков асинхронной машины, представленный в разделе **Machines** библиотеки **SimPowerSystems Specialized Technology**.

2.3. Моделирование асинхронного двигателя с короткозамкнутым ротором

Для моделирования асинхронной машины (Рис. 2.40) воспользуемся следующими блоками: (*подраздел Electrical Sources*) **Three-Phase Source**; (*Measurements*) **Three-Phase V-I Measurement**; (*Elements*) **Three-Phase Breaker**; (*Machines*) **Asynchronous Machine SI Units**; (*Simulink*) **Step**, и параметрами из *Примера 1.7*. Сохраним модель в файле **example_2_8**.

В окне задания параметров трехфазного источника (**Three-Phase Source**) введем значения напряжения 380 В , частоты 50 Гц . Сопротивление и индуктивность источника примем равными нулю. Соединение обмоток источника – Y_n . Базисное напряжение – 380 В .

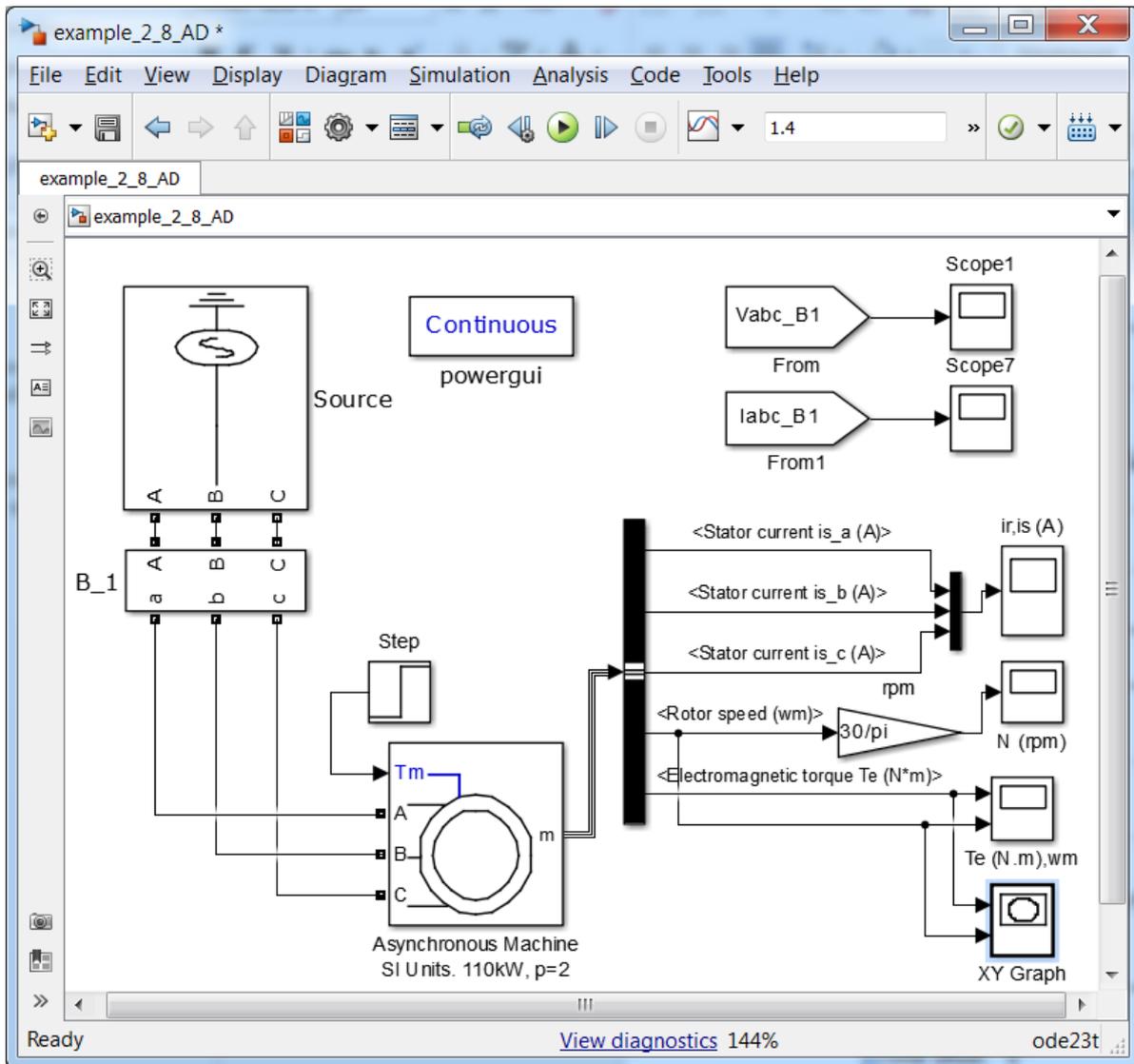


Рис. 2.40

В окне параметров выключателя **Three-Phase Breaker** время срабатывания задано равное нулю. Остальные параметры – по умолчанию.

Первая вкладка **Configuration** окна параметров блоком **Asynchronous Machine SI Units** (Асинхронная машина с параметрами в единицах СИ) представлена на рис. 2.41.

На ней в выпадающих списках указаны:

- **Preset model:** (*установленные модели*). В выпадающем списке можно выбрать асинхронный двигатель для загрузки его параметров;

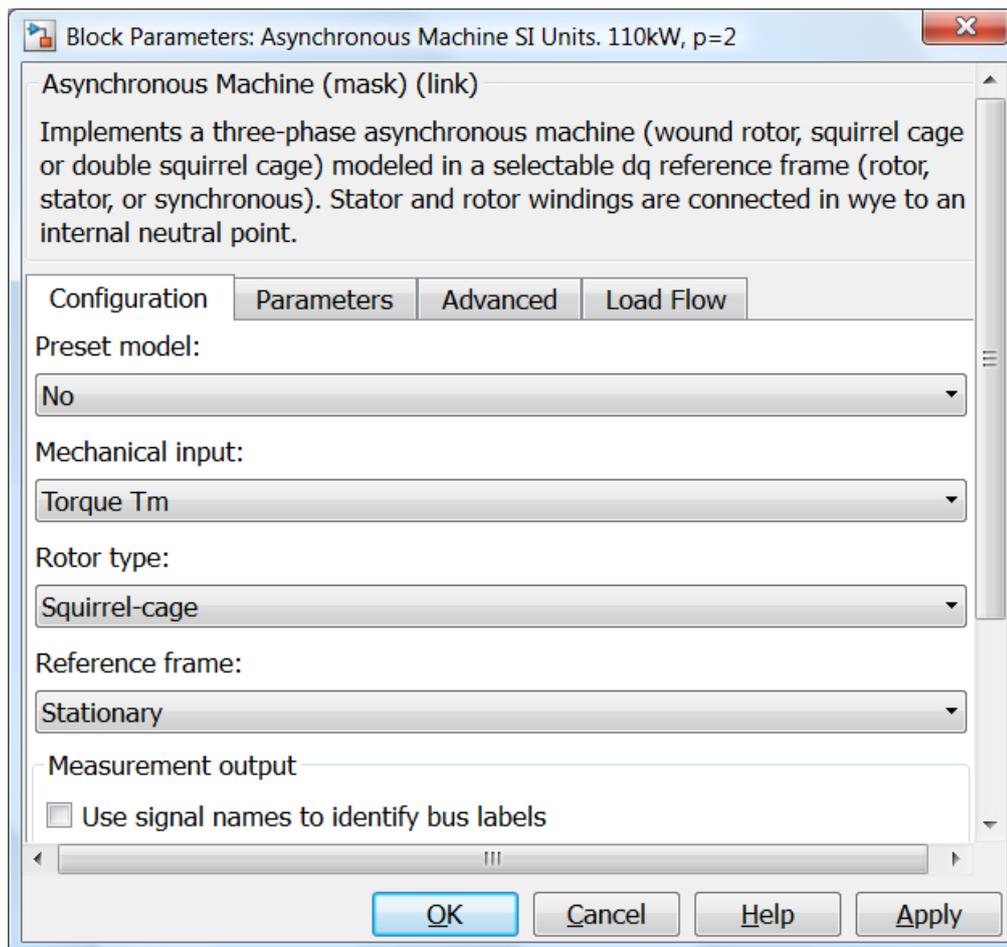


Рис. 2.41

- **Mechanical input:** (*механический вход*). В зависимости от выбора на механический вход можно подать **Torque Tm** (*момент*), **Speed w** (*скорость*) или создать порт механического вращения, для взаимодействия с механическим валом библиотеки **Simscape**;
- **Rotor type:** (тип ротора). В выпадающем списке можно выбрать тип ротора: фазный, "беличья клетка" (короткозамкнутый ротор) или ротор с двойной "беличьей клеткой";
- **Reference frame.** Система координат, которая принимается в математической модели машины: неподвижная относительно ротора; неподвижная относительно статора, вращающаяся синхронно с полем.

Во вкладке **Parameters** (рис. 2.42) представлены окна, в которые необходимо ввести соответствующие параметры:

1. **Nominal power, voltage (line-line), and frequency [Pn(VA), Vn(Vrms), fn(Hz)]** – номинальная активная мощность двигателя, линейное напряжение и частота.

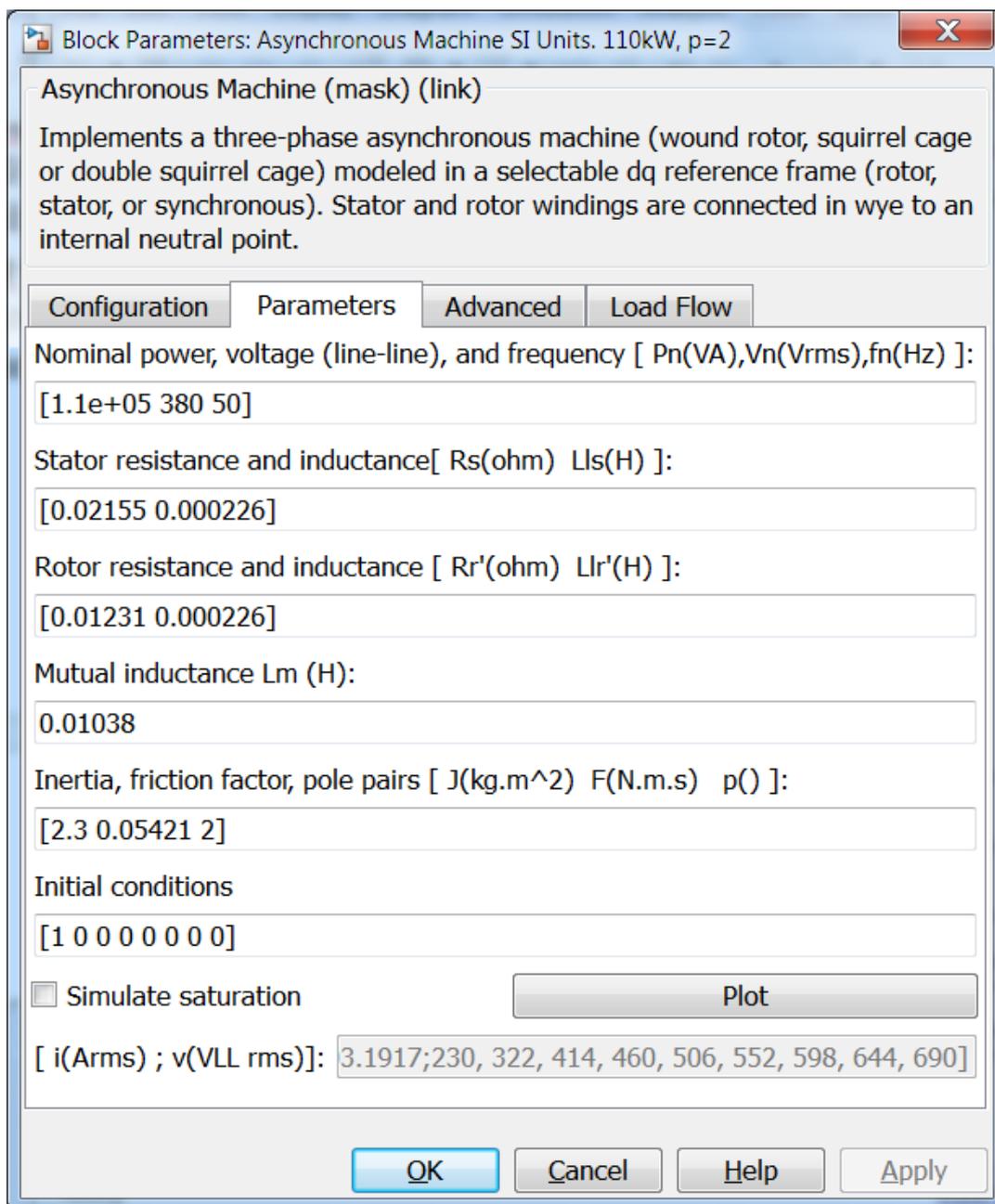


Рис. 2.42

Следует отметить, что в отечественной литературе под размерностью **VA** подразумевают полную, а не активную мощность.

2. **Stator resistance and inductance [$R_s(\text{ohm})$ $L_{ls}(\text{H})$]:** – активное сопротивление и индуктивность рассеяния обмотки статора.
3. **Rotor resistance and inductance [$R_r'(\text{ohm})$ $L_{lr}'(\text{H})$]:** – приведенные к обмотке статора активное сопротивление и индуктивность рассеяния обмотки ротора.

4. **Mutual inductance L_m (H):** – взаимная индуктивность обмоток, расположенных на статоре и роторе.
5. **Inertia, friction factor, pole pairs [$J(\text{kg.m}^2)$ $F(\text{N.m.s})$ $p()$]:** – момент инерции, коэффициент трения, число пар полюсов.
6. **Initial conditions** – начальные условия переменных (скольжение, электрический угол, амплитуды токов трех фаз статора, соответствующие фазы этих токов).
7. **Simulate saturation** – моделирование насыщения.
8. **Plot** – построение кривой намагничивания. Во вкладке **Advanced** (модифицированный) предложена дискретная модель (**Discrete solver model**) и возможность выбора соответствующего метода расчета.

В окна блока **Step M_nom** заносим следующие значения: **Step time: 1; Initial value: 0; Final value: 706.4**. Остальные – по умолчанию.

Результаты моделирования переходного процесса при пуске и набросе номинальной нагрузки представлены на рис. 2.43. Они совпали с результатами, которые были получены в *Примере 1.7* (рис. 1.18).

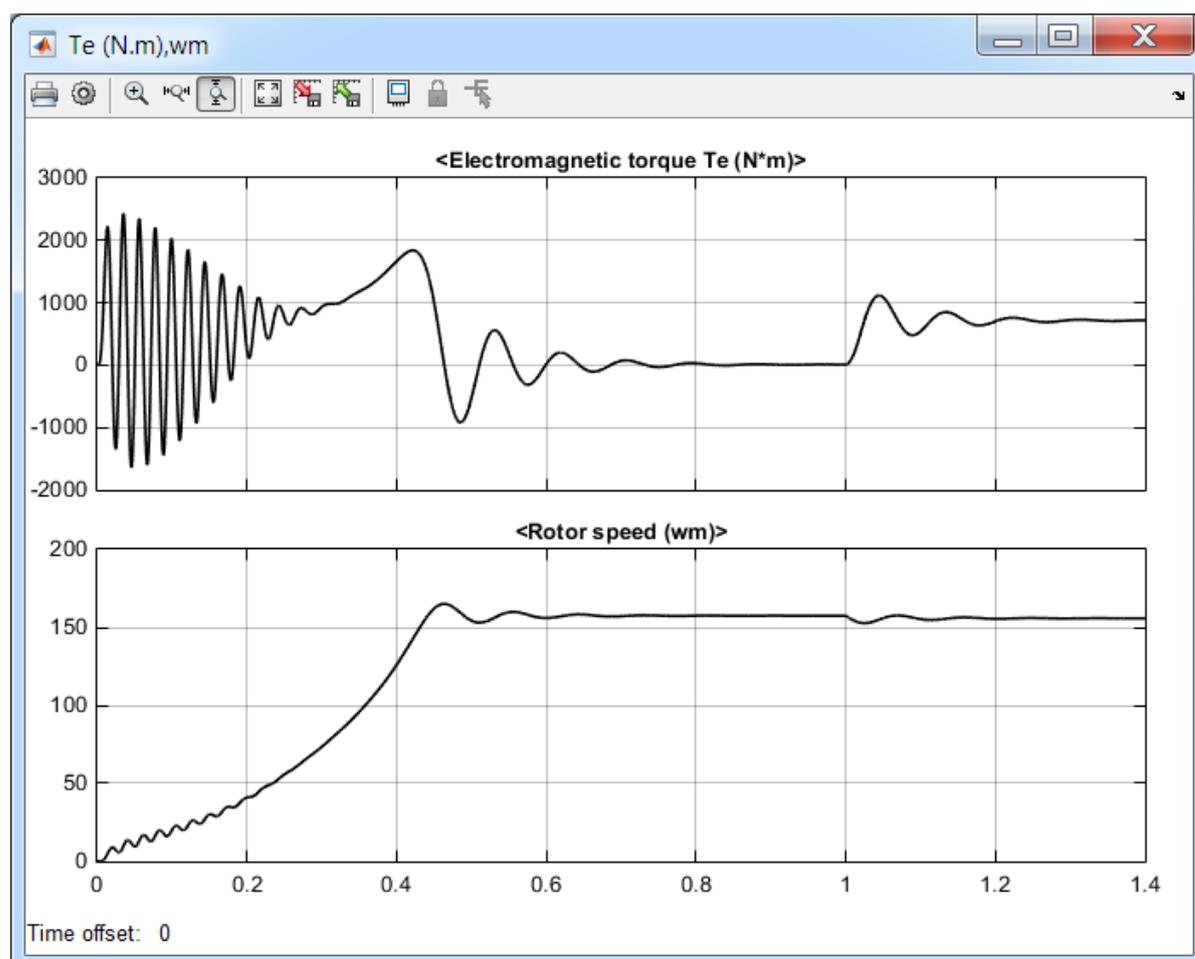


Рис. 2.42

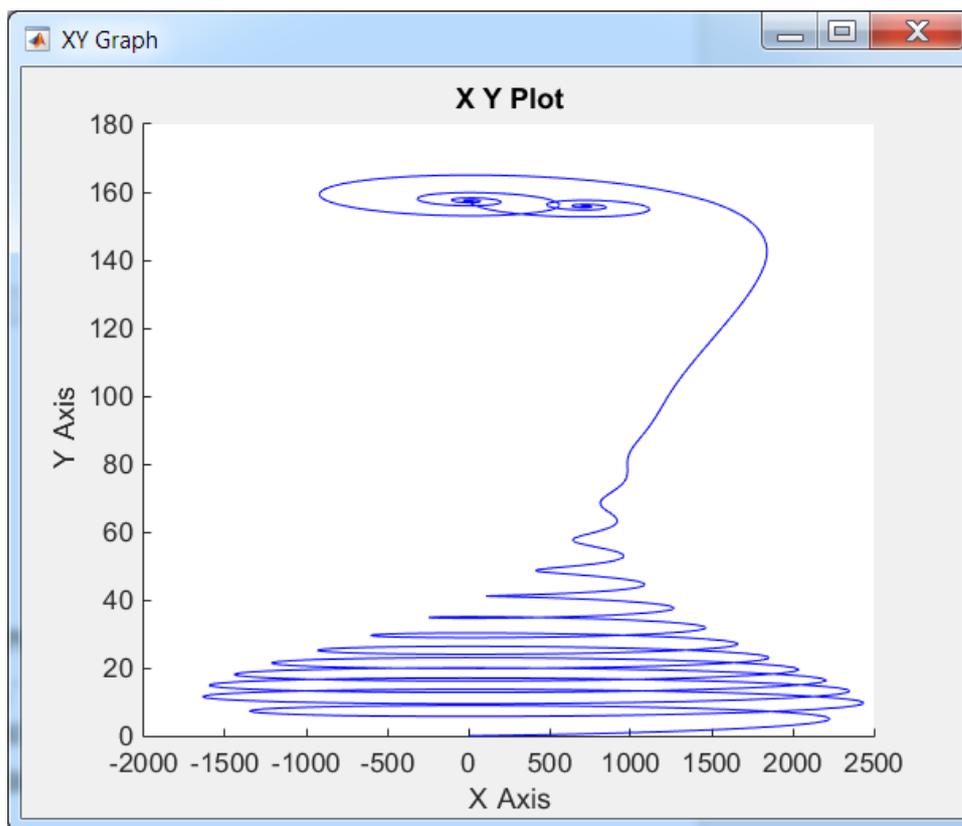


Рис. 2.43

На рис. 2.43 представлена динамическая механическая характеристика асинхронного двигателя, выведенная на графопостроитель (**XY Graph**). Предварительно его оси необходимо задать в окне параметров блока, оценив максимальные и минимальные значения по рис. 2.42.

Процесс изменения токов в фазах обмотки статора представлен на рис. 2.44. По этим показаниям можно оценить и кратность пускового тока ($k = I_{\text{пуск}} / I_{\text{ном}} \approx 6,95$).

2.4. Моделирование трансформаторных подстанций 10/0,4 кВ

Трансформаторные подстанции являются основным звеном системы электроснабжения предприятий. В связи с этим моделирование переходных процессов, протекающих в этих подстанциях, является важной задачей. Результаты моделирования позволят правильно выбрать оборудование, оценить влияние различных переходных процессов на качество электроснабжения, выбрать компенсирующие устройства и т.д.

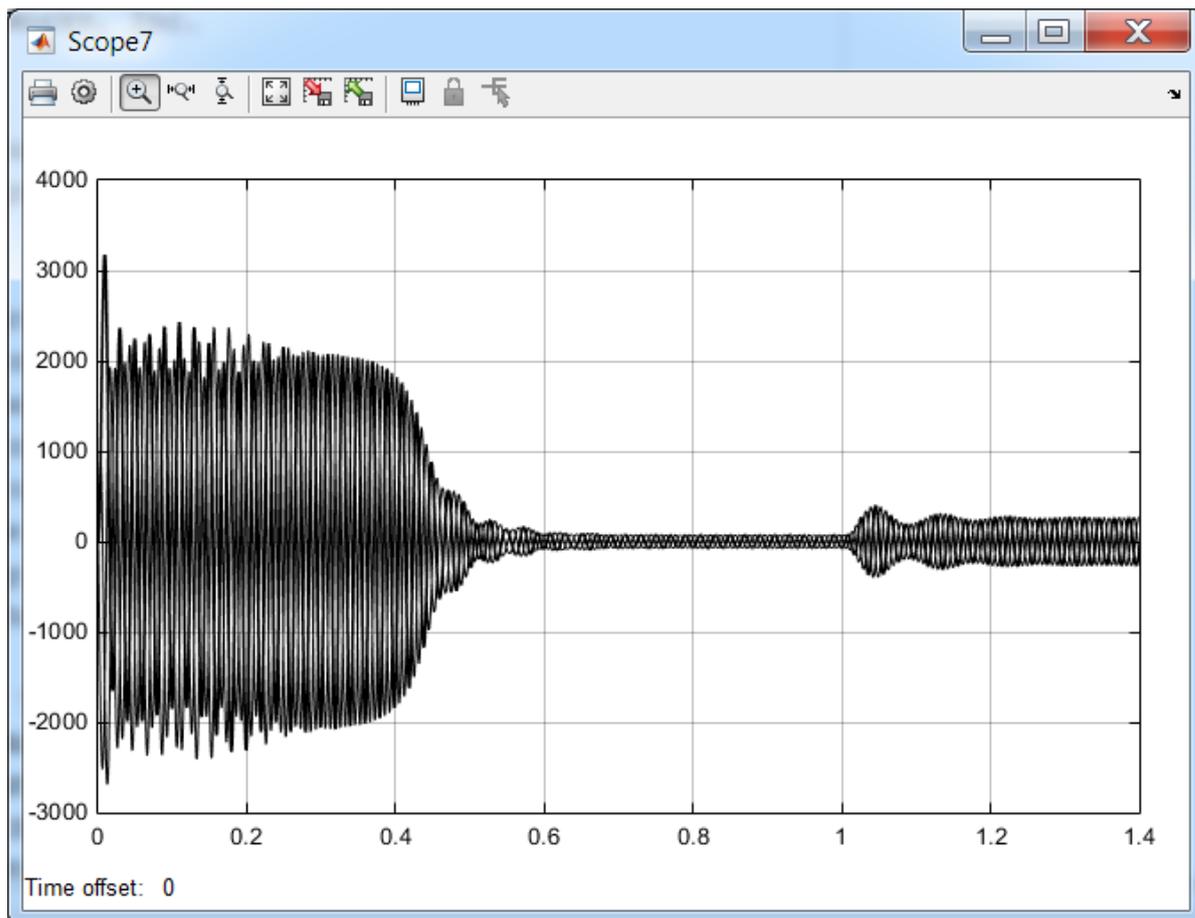


Рис. 2.44

2.5.1. Однотрансформаторная подстанция

В качестве примера рассмотрим моделирование переходных процессов в однотрансформаторной подстанции.

Пример 2.9.

Рассчитать переходные процессы при коротких замыканиях, набросе и сбросе нагрузки, подключении асинхронного двигателя, трехфазном коротком замыкании в однотрансформаторной подстанции. Модель цеха показана на рис. 2.44.

Решение

Пусть параметры подстанции имеют следующие значения:

- 1) Трехфазный источник напряжения: первичное линейное напряжение – 10 кВ; частота напряжения – 50 Гц; соединение фаз обмотки источника – Y_n .
- 2) Кабельная линия **L_1** (блок **Three-Phase Series RLC Branch**): активное сопротивление – 0,2 Ом; индуктивность – 2,48E-5 Гн.
- 3) Параметры данного трансформатора аналогичны параметрам трансформатора из *Примера 2.7* (Рис. 2.32, стр. 108).

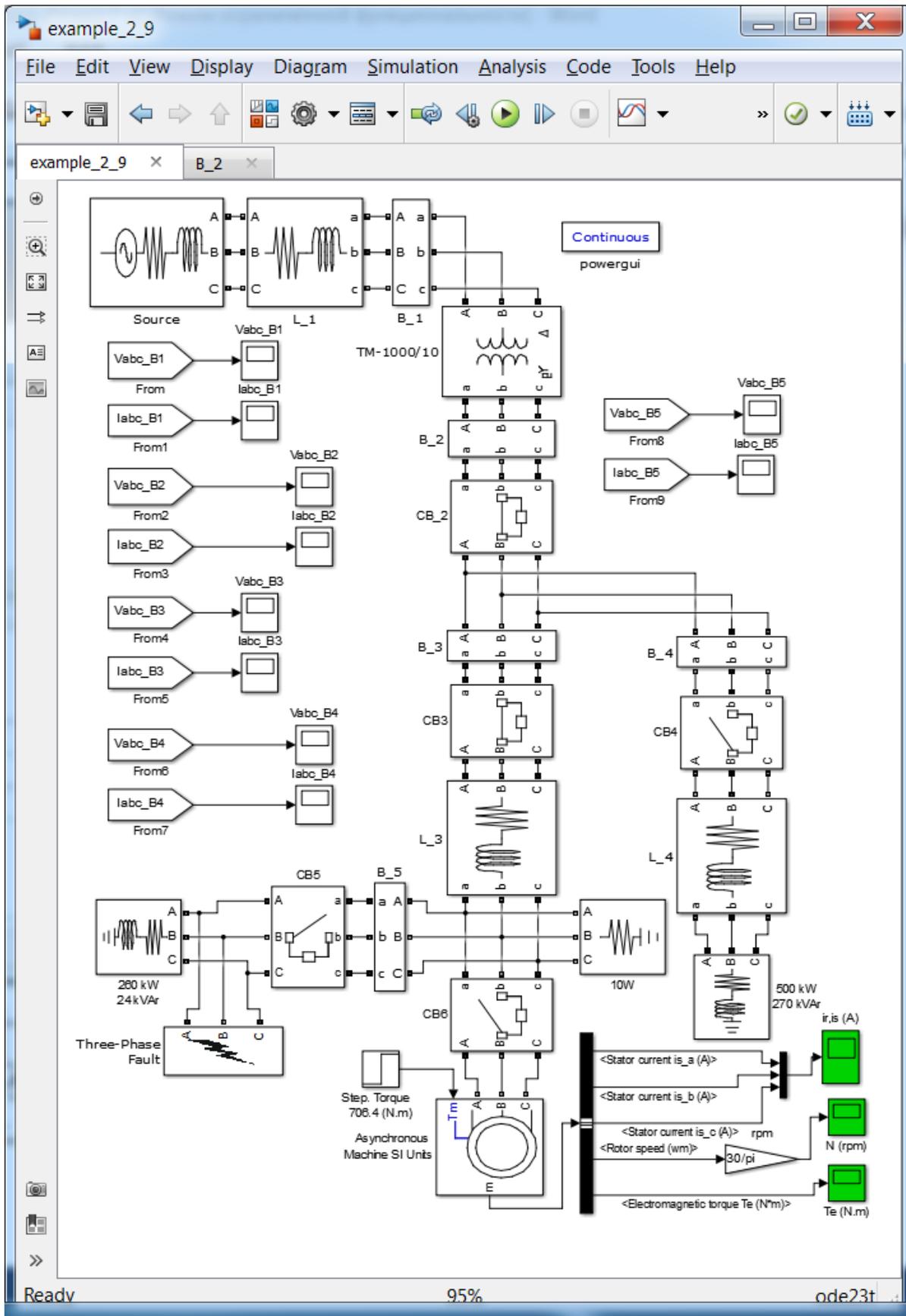


Рис. 2.44

- 4) Блоки **B2, B3, B4, B5** это блоки измерения трехфазных напряжений и токов (**Three-Phase V-I Measurement**).
- 5) Выключатель **CB2** (блок **Three-Phase Breaker**): активное сопротивление – $0,00014 \text{ Ом}$; время отключения (**Switching times (s):**)– [10.5], т.е. при данном времени расчета [125/50] этот выключатель будет находиться в замкнутом состоянии, время отключения (**Switching times (s):**)– [10.5].
- 6) Выключатель **CB3**: активное сопротивление – $0,00041 \text{ Ом}$; время отключения – [10.5].
- 7) Кабельная линия **L_3**: активное сопротивление – $0,0032 \text{ Ом}$; индуктивность – $1,8\text{E-}6 \text{ Гн}$.
- 8) Выключатель **CB5**: активное сопротивление – $0,00041 \text{ Ом}$; время включения и отключения [5/50 101/50], т.е. через пять периодов нагрузка по пункту 9 подключается к трансформатору, а через 101 период отключается, при этом отключается и ток короткого замыкания, который создается с помощью блока по пункту 10.
- 9) Последовательная трехфазная RLC нагрузка **260 kW 24 kVAr** (блок **Three-Phase Series RLC Load**): активная мощность – 260 кВт ; реактивная мощность индуктивного характера – 24 ВАр .
- 10) Блок **Three-Phase Fault** (*Блок трехфазный замыканий*): трехфазное короткое замыкание – отмечены флажками три фазы; установлено время короткого замыкания [100/50 1010/50]; сопротивление дуги (**Fault resistance Ron (Ohm):**) – $0,001 \text{ Ом}$.
- 11) Выключатель **CB6**: активное сопротивление – $0,0011 \text{ Ом}$, подключает асинхронный двигатель через десять периодов – [10/50].
- 12) Параметры асинхронного двигателя совпадают с параметрами двигателя из *Примера 2.8 (Рис. 2.42, стр. 118)*.
- 13) Блок нагрузки **10W (Three-Phase Series RLC Load)** необходим для корректной работы модели при отсутствии нагрузки в конце линии.
- 14) Выключатель **CB4** с активным сопротивлением $0,00025 \text{ Ом}$ подключает активно-индуктивную нагрузку (пункт 15) через семьдесят пять периодов– [75/50].
- 15) Кабельная линия **L_4**: активное сопротивление – $0,0016 \text{ Ом}$; индуктивность – $1.11\text{E-}6 \text{ Гн}$.
- 16) Последовательная трехфазная RLC нагрузка **500 kW 270 kVAr** (блок **Three-Phase Series RLC Load**): активная мощность – 500 кВт ; реактивная мощность индуктивного характера – 270 ВАр .

В данной модели время расчета составляет сто двадцать пять периодов 125/50. Выключатели **CB2** и **CB3** во время расчета находятся в

замкнутом состоянии. В модели задана следующая последовательность коммутаций выключателей и блока коротких замыканий:

- 1) пять периодов трансформатор работает на холостом ходу [5/50];
- 2) в момент времени [5/50] выключатель **CB5** подключает активно-индуктивную нагрузку (блок **260 kW 24 kVAr**);
- 3) через десять периодов [10/50] выключатель **CB6** подключает асинхронный двигатель (**Asynchronous Machine SI Units**), мощностью **110 кВт**;
- 4) через пятьдесят периодов [50/50] после разгона двигателя на холостом ходу на него набрасывают номинальную нагрузку (блок **Step. Torque 706.4 (N.m)**);
- 5) через семьдесят пять периодов выключатель **CB4** подключает ещё одну активно-индуктивную нагрузку (блок **500 kW 270 kVAr**);
- 6) через сто периодов [100/50] с помощью блока **Three-Phase Fault** осуществляется трехфазное короткое замыкание;
- 7) через сто один период [101/50] выключатель **CB5** отключает короткое замыкание и нагрузку.

На рис. 2.45 представлены результаты расчета токов при моделировании перечисленных выше семи переходных процессов (с учетом холостого хода).

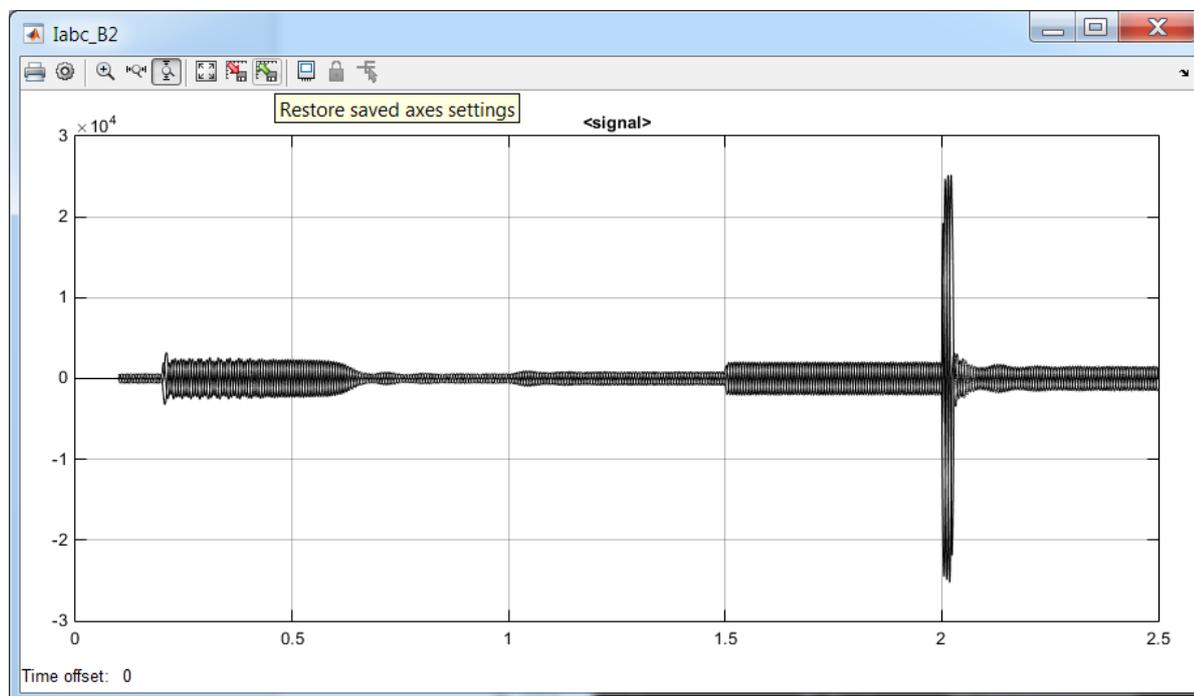


Рис. 2.45

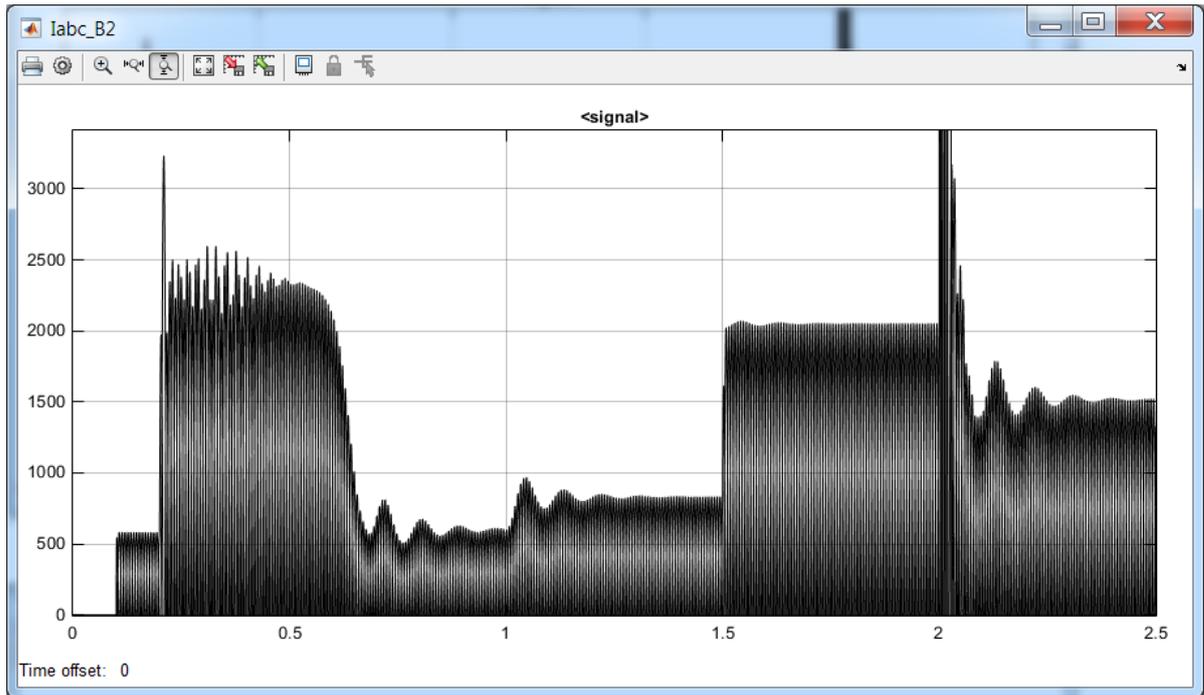


Рис. 2.46

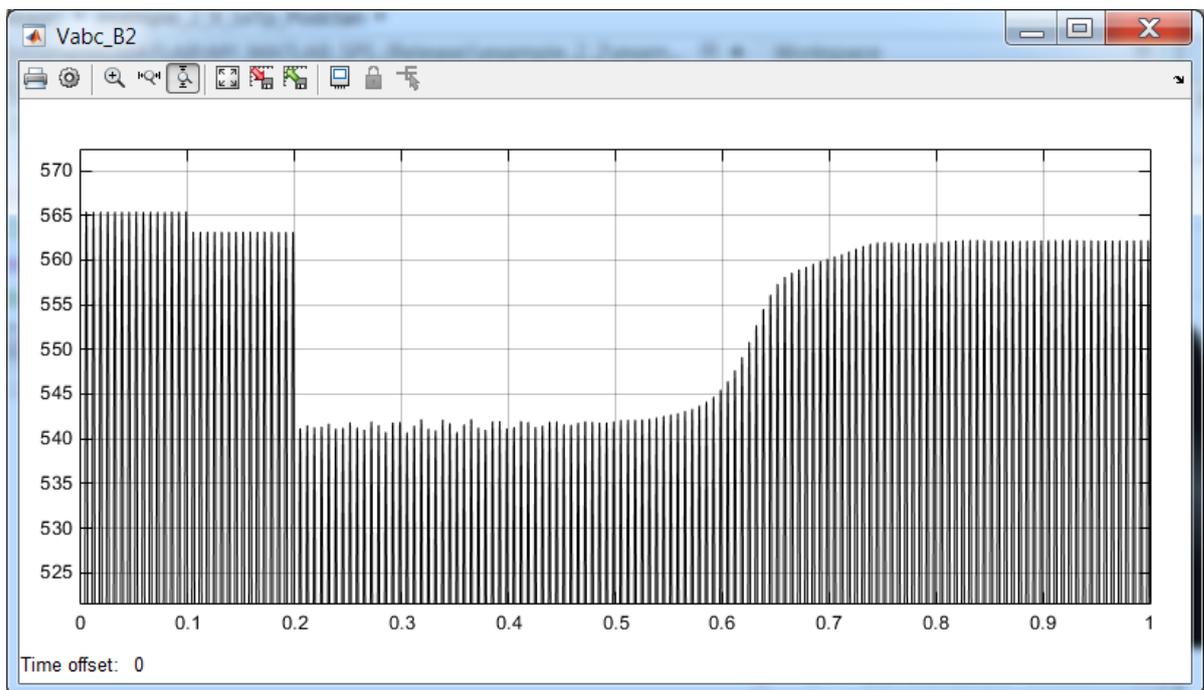


Рис. 2.47

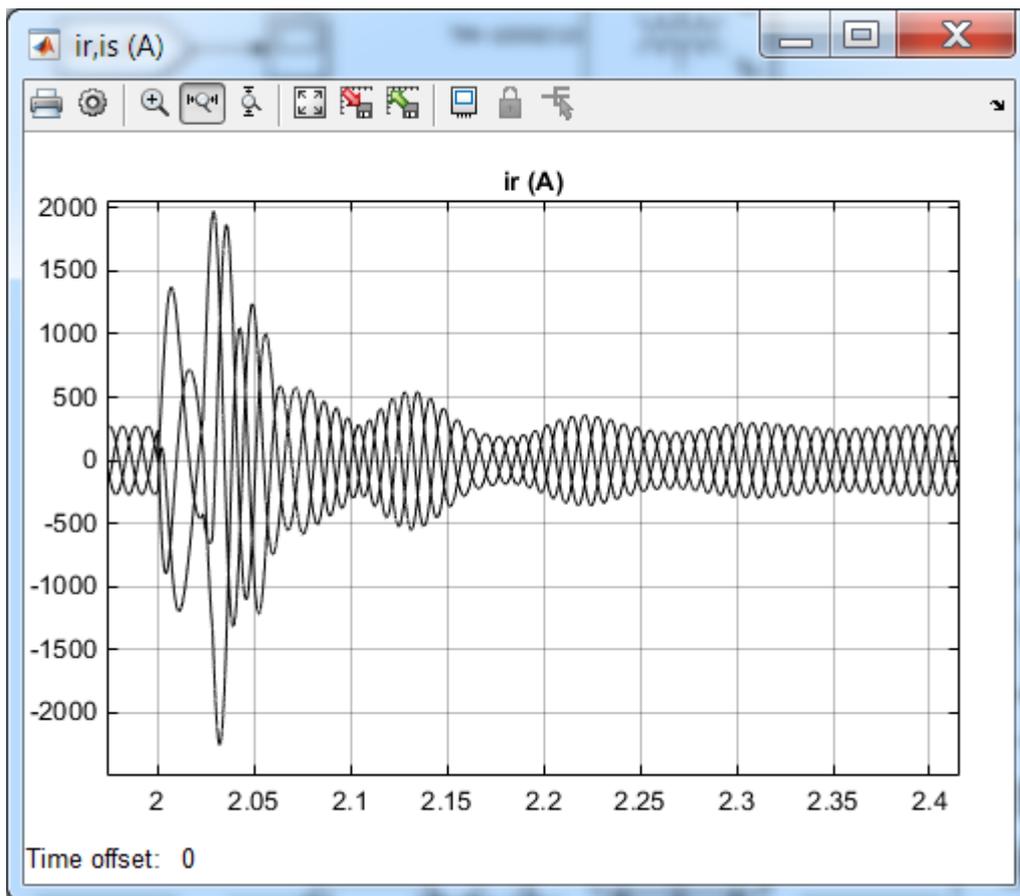


Рис. 2.48

В связи с тем, что на этом рисунке показаны не только токи нагрузки, но и токи трехфазного короткого замыкания, увидеть особенности переходных процессов при набросе нагрузки можно только изменив масштаб. На рис. 2.46 показана положительная часть синусоидальных кривых токов, которая дает представление о переходных процессах.

Изменение токов в трансформаторе приводит к соответствующим изменениям напряжений (Рис. 2.47). Наибольшее снижение напряжения, если не считать короткое замыкание, наблюдается при пуске асинхронного двигателя.

Следует отметить, что трехфазное короткое замыкание оказывает негативное влияние на асинхронный двигатель. В частности, наблюдаются токи, соизмеримые с пусковыми токами (рис. 2.48).

2.5.1. Двухтрансформаторная подстанция

В системах электроснабжения промышленных предприятий наибольшее распространение получили двухтрансформаторные подстанции. Основным достоинством таких подстанций является

возможность резервирования электроснабжения. В нормальных режимах для уменьшения токов короткого замыкания два трансформатора работают отдельно, т.е. секционный выключатель разомкнут. В случае исчезновения напряжения на шинах потребителя срабатывает система автоматического ввода резерва. Она отключает вводной выключатель секции без напряжения и только потом включает секционный выключатель, т.е. подключает нагрузку к оставшемуся в работе трансформатору.

При восстановлении напряжения схема возвращается в исходное состояние, при этом некоторое время трансформаторы работают параллельно. В свою очередь, для включения трансформатора необходимо выполнить ряд условий, а именно: трансформаторы должны принадлежать одной группе соединения; иметь одинаковые коэффициенты трансформации; иметь одинаковые напряжения короткого замыкания.

Таким образом, создание модели для изучения особенностей параллельной работы трансформаторов представляет определенный интерес. Рассмотрим пример, который посвящен изучению параллельной работы трансформаторов при неравенстве коэффициентов трансформации.

Пример 2.10.

При параллельной работе двух трансформаторов найти: уравнивающий ток, если вторичные напряжения соответственно равны 400 В и 380 В; токи в обмотках при номинальной активной нагрузке.

Решение

Для решения поставленных задач соберем модель, которая представлена на рис. 2.49 (силовая часть) и рис. 2.50 (измерительная часть) [11]. Параметры трехфазных источников и трансформаторов подстанции точно такие же, как у источника и трансформатора из *Примера 2.9* и *Примера 2.7* (Рис. 2.32, стр. 108). Активное сопротивление всех выключателей равно 0,00014 Ом.

Во всех выключателях задаем время коммутации равное 15 с. В выключателях СВ_1 и СВ_2 устанавливаем начальное состояние – **closed**, а в остальных – **open**. Время расчета задаем равным 0,024 с. Выбираем решатель **ode23tb(stiff/Tr-BDf2)**, задаем относительную точность **1e-6** и запускаем модель. На рис. 2.50 представлены результаты расчета работы трансформаторов на холостом ходу.

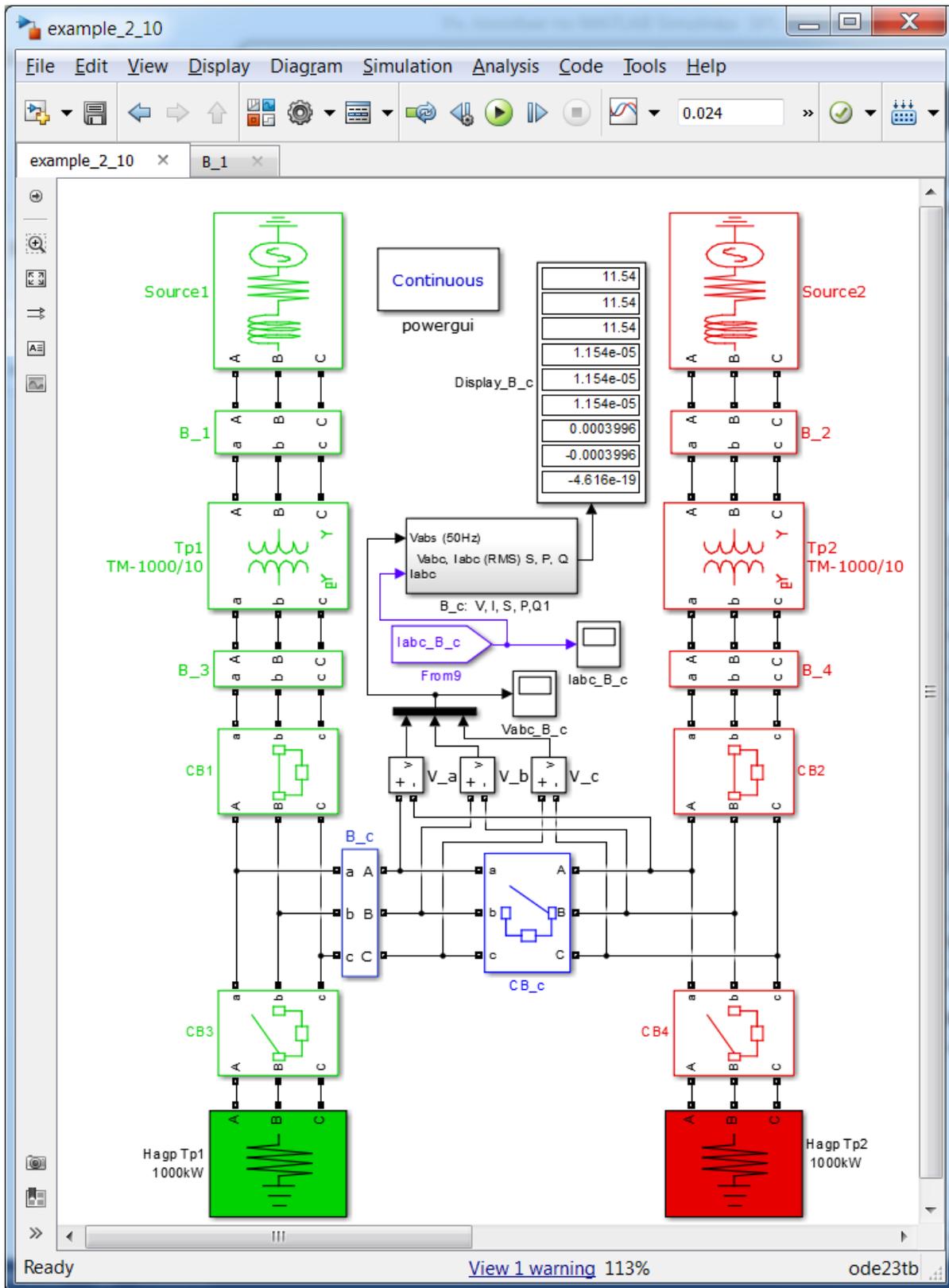


Рис. 2.49

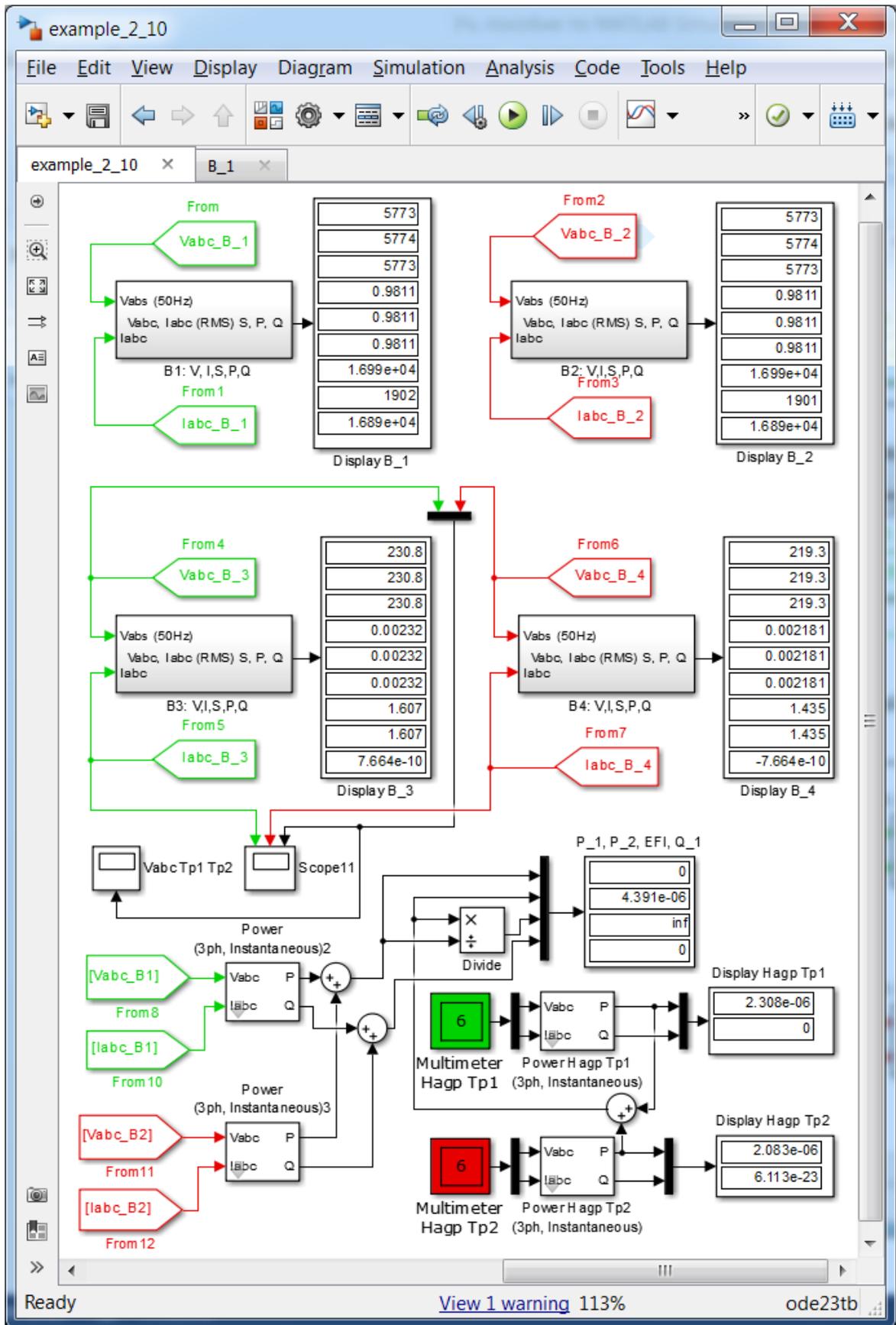


Рис. 2.50

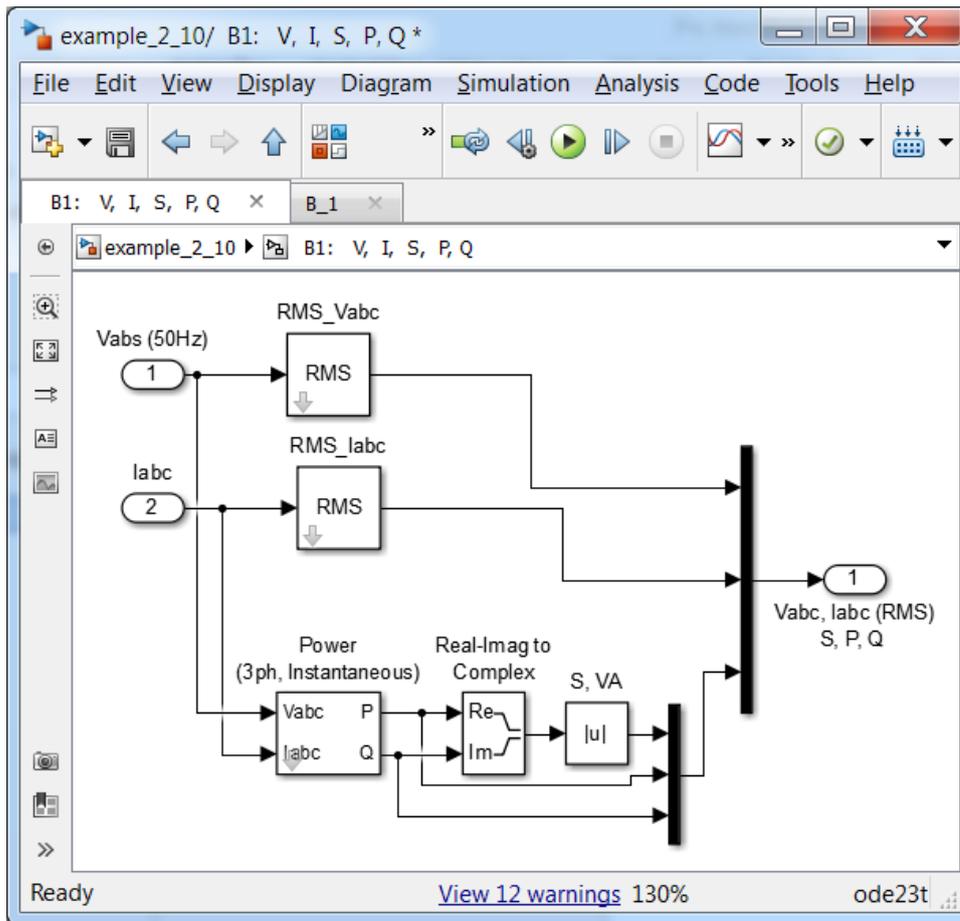


Рис. 2.51

Показания **Display B_1** и **Display B_2** по активной мощности (*предпоследнее окно*) свидетельствуют о хорошей сходимости, поскольку расчетные значения потерь холостого хода трансформаторов соответственно равны **1902 Wm** и **1901 Wm**, а паспортное значение – **1900 Wm**. Согласно данным, показанным на дисплеях **Display B_3** и **Display B_4**, вторичные напряжения трансформаторов отличаются и равны **230.8** и **219.3**.

Следует отметить, что используемые в модели четыре подсистемы для вывода напряжений, токов и мощностей **B1: V,I,S,P,Q**, **B2: V,I,S,P,Q**, **B3: V,I,S,P,Q**, **B4: V,I,S,P,Q** и **B_c: V,I,S,P,Q** имеют в своем составе блоки, представленные на рис. 2.51.

Для определения уравнильных токов переводим выключатели **CB_1** и **CB_2** в состояние **closed**, секционный выключатель **CB_c** в состояние **open**. Задаем время коммутации последнего **[0.025 0.16]**. Время срабатывания выключателей **CB_3** и **CB_4** задаем равным **[0.16]**, а первоначальное состояние **open**. Время расчета увеличиваем до **0.15** и после запуска модели получаем значения, представленные на рис. 2.52.

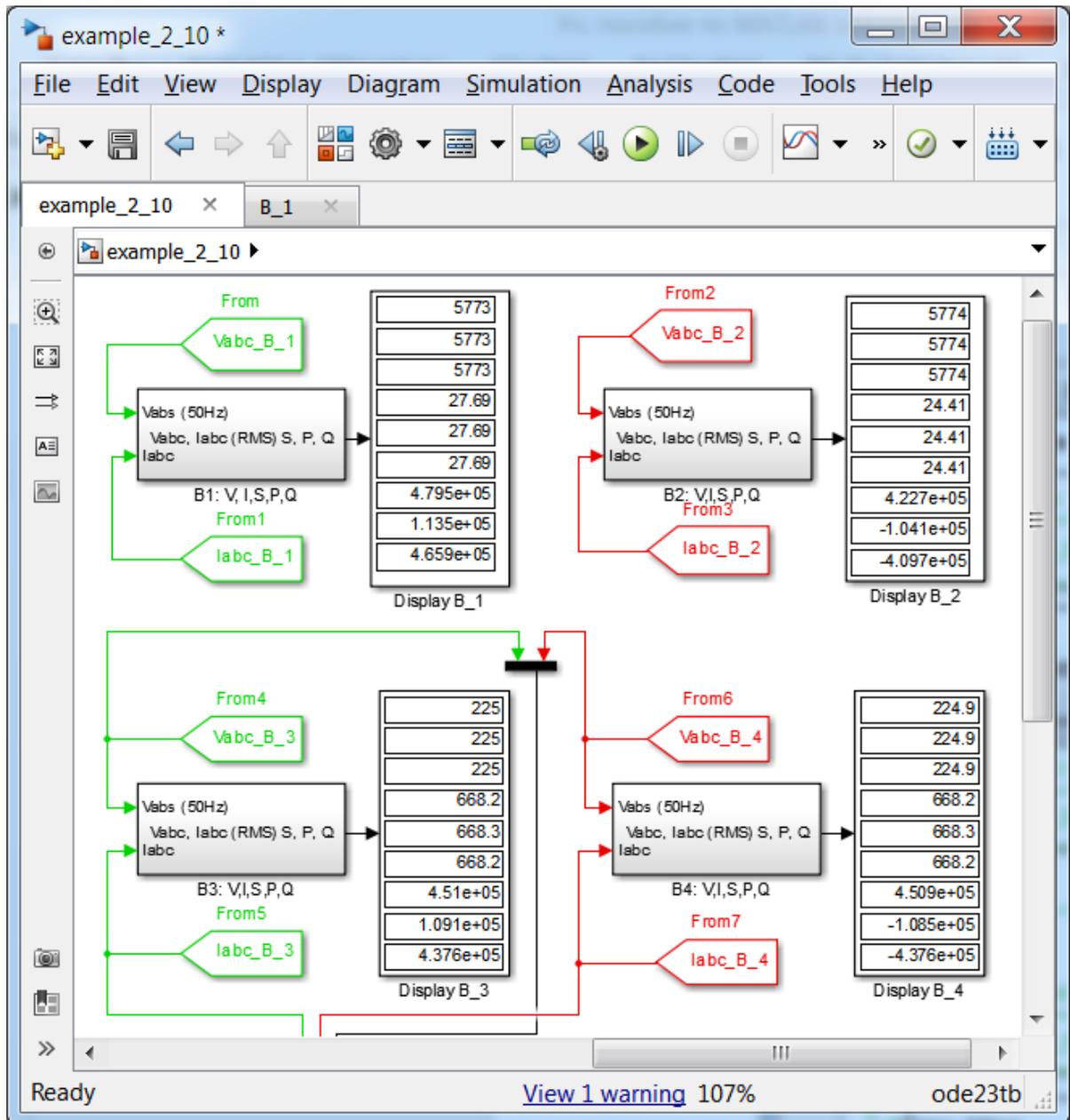


Рис. 2.52

Уравнительные токи, протекающие во вторичных обмотках трансформаторов, одинаковы и равны **668.2 А**. Вторичные напряжения практически равны **225 В** и **224.9 В**. Токи в первичных обмотках трансформаторов отличаются, по всей видимости, на величину намагничивающего тока. При этом потери в трансформаторах составили величину $1.135e+5 - 1.041e+5 = 9400 \text{ Вт}$, что превышает общие потери трансформаторов на холстом ходу, т.е. **3800 Вт**.

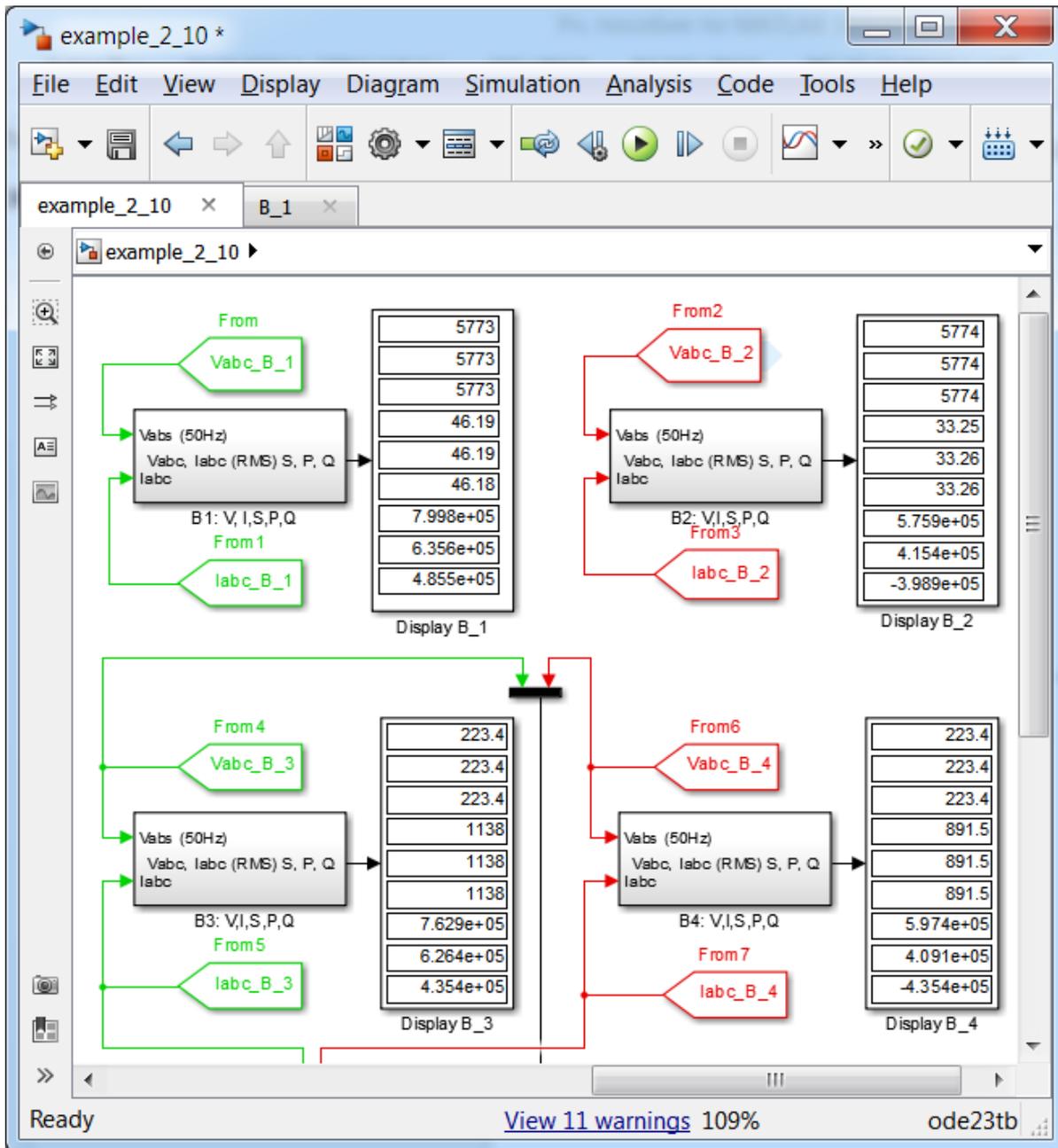


Рис. 2.53

Для определения токов при параллельной работе трансформаторов и номинальной нагрузке в выключателях **СВ_3** и **СВ_4** изменяем время коммутации на **[0.03]**. В результате получаем следующие значения токов: **46,19 А**, **33,26 А** – токи в первичных обмотках трансформаторов; **1138 А**, **891,5 А** – токи во вторичных обмотках трансформаторов (рис. 2.53).

Достаточно интересная картина (**Scope11**) переходных процессов получается при работе трансформаторов на холостом ходу, включении ($t = 0.02$ с) и отключении ($t = 0.06$ с) секционного выключателя, при включении ($t = 0.08$ с) секционного выключателя с последующим

набросом ($t = 0.1$ с) нагрузки (рис. 2.54). На отрезке от $t = 0.02$ с до $t = 0.06$ с видно, что уравнивающие токи в трансформаторах равны и находятся в противофазе. При набросе нагрузки ($t = 0.1$ с) происходит её неравномерное распределение между трансформаторами. Больше нагружается трансформатор с большим вторичным напряжением, т.е. с меньшим коэффициентом трансформации.

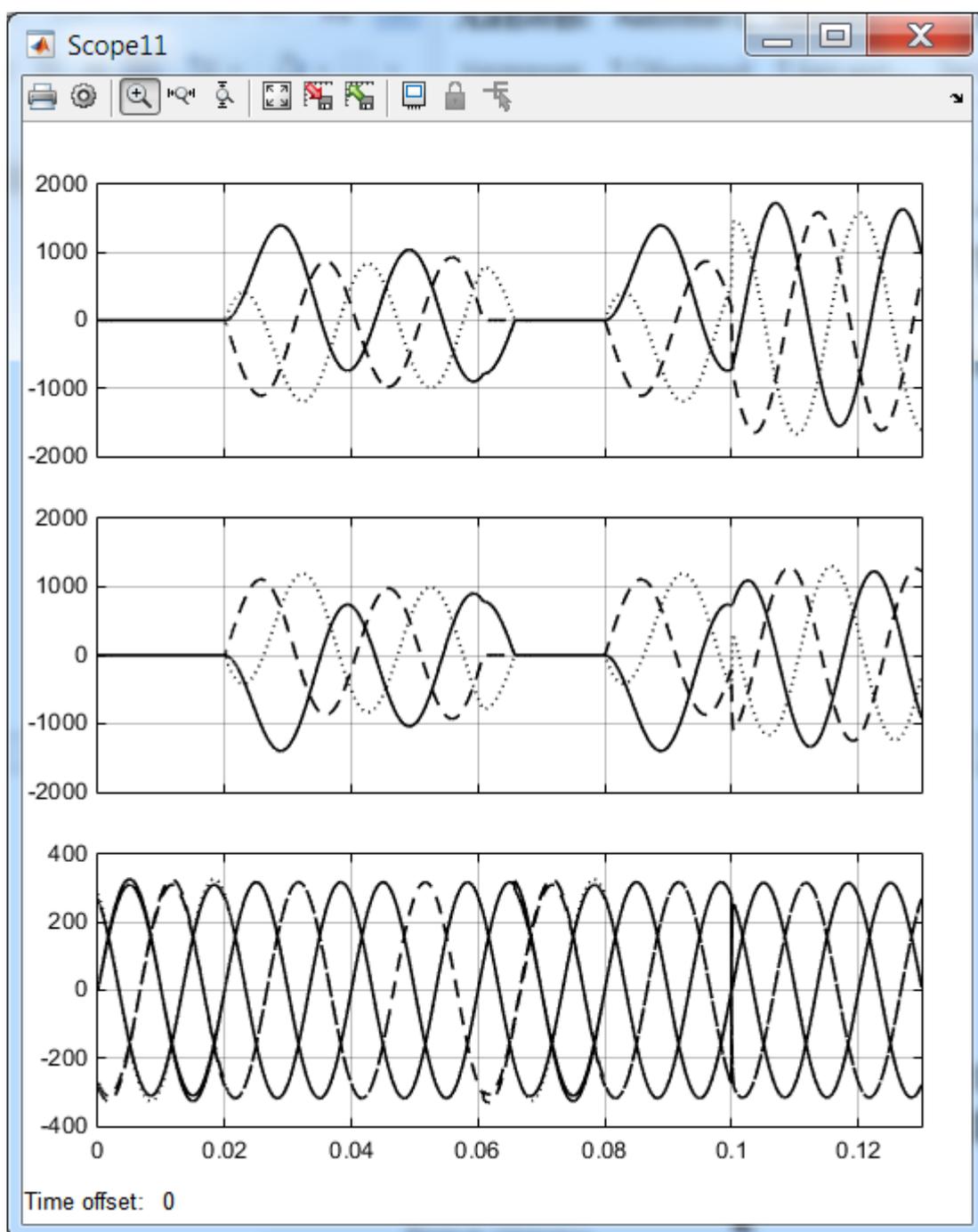


Рис. 2.54

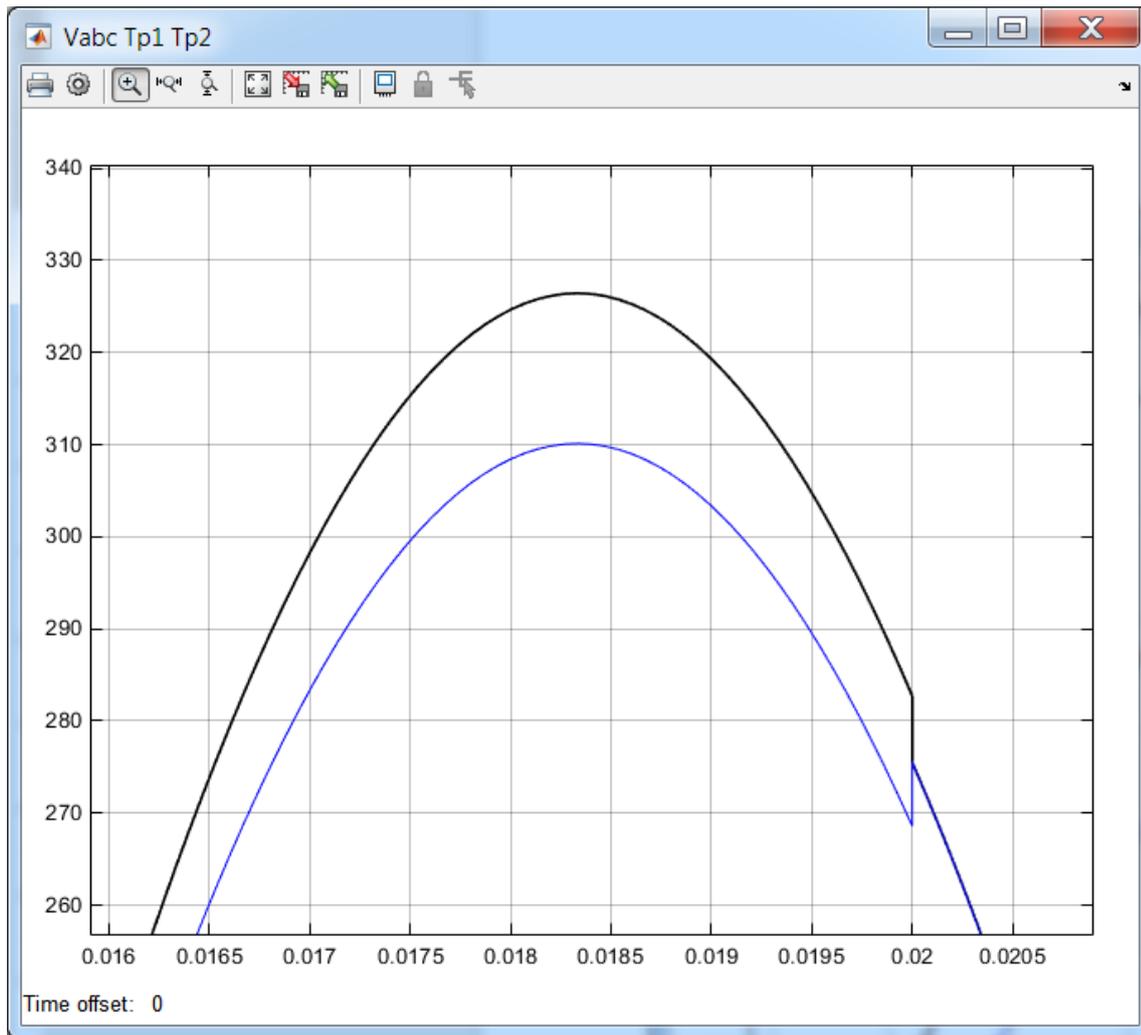


Рис. 2.55

В более крупном масштабе, на рис. 2.55 представлен процесс выравнивания напряжений на уровне кривых напряжений. Здесь верхняя кривая это часть синусоиды вторичного напряжения первого трансформатора, а другая кривая – второго трансформатора. При $t = 0.02$ с происходит включение секционного выключателя и выравнивание напряжений.

Для получения таких кривых задаем время расчета **0.13 с** и следующие интервалы коммутации выключателей: **СВ_с – [0.02 0.06 0.08]**; **СВ_3 и СВ_4 – [0.1]**.

Приведенные в учебном пособии примеры продемонстрировали только небольшую часть тех возможностей, которые предоставляет **MATLAB**. Дальнейшее её изучение позволит существенно сократить время расчета курсовых и дипломных проектов и, шаг за шагом, откроет вам огромные возможности такого замечательного продукта как **MATLAB** и её расширения **Simulink** и **SimPowerSystems**.

Список литературы

1. **Курбатов, Е. А.** *MATLAB 7. Самоучитель.* –М.: : Издательский дом "Вильямс", 2006.
2. **Черных, И. В.** *Моделирование электротехнических устройств в MATLAB, SimPowerSystems и Simulink.* –М : ДМК Пресс; Питер, 2008.
3. **Дьяконов, В. П. и Пеньков, А. А.** *MATLAB и Simulink в электроэнергетике. Справочник.* –М : Горячая линия-Телеком, 2009.
4. **Беспалов, В. Я. и Котеленец, Н. Ф.** *Электрические машины.* –М : Издательский центр "Академия", 2013.
5. **Копылов, И. П.** *Математическое моделирование электрических машин.* –М.: : Высш. шк., 2001.
6. *youtube.com.* [В Интернете] [Цитировано: 21 12 2014 г.]
https://www.youtube.com/watch?v=-QtVdcS2Lko&list=PLmu_y3-DV2_mCc5DAIxZnyQ7itvu1RM9H.
7. **Герман-Галкин, С. Г.** *Matlab & Simulink. Проектирование мехатронных систем на ПК.* – СПб. : КОРОНА-Век, 2008.
8. **Дьяконов, В. П.** *Simulink 5/6/7: Самоучитель.* – М. : ДМК-Пресс, 2008.
9. *SimPowerSystems . MATLAB.Exponenta.* [В Интернете] [Цитировано: 15 декабрь 2014 г.] <http://matlab.exponenta.ru/simpower/book1/index.php>.
10. *Simscape. MathWorks Центр компетенций.* [В Интернете] MathWorks . [Цитировано: 28 декабрь 2014 г.] <http://matlab.ru/products/simscape>.
11. **Джэндубаев, Эдуард Абрек-Заурович.** Разработка модели цеховой двухтрансформаторной подстанции в среде MATLAB и её расширениях Simulink и SimPowerSystems. *Известия Северо-Кавказской государственной гуманитарно-технологической академии.* РИЦ СевКавГГТА, 2014 г., № 1.

ДЖЕНДУБАЕВ Абрек-Заур Рауфович,
АЛИЕВ Исмаил Ибрагимович

MATLAB, Simulink и SimPowerSystems в электроэнергетике

**Учебное пособие
для студентов, обучающихся по направлению
подготовки 140400.62 "Электроэнергетика и электротехника",
профиль "Электроснабжение"**

Печатается в редакции авторов

Корректор
Редактор

Сдано в набор
Формат 60x84/16
Бумага офсетная.
Печать офсетная.
Усл. печ. л.
Заказ №
Тираж

Оригинал-макет подготовлен в Библиотечно-издательском
центре СевКавГГТА

369000, г. Черкесск, ул. Ставропольская, 36

