

**А.С. Красько**

# **ОСНОВЫ КОМПЬЮТЕРНОГО ПРОЕКТИРОВАНИЯ И МОДЕЛИРОВАНИЯ РАДИОЭЛЕКТРОННЫХ СРЕДСТВ**

*Учебное методическое пособие*



**ТОМСК – 2012**

Министерство образования и науки Российской Федерации

**ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ  
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)**

**Кафедра радиоэлектроники и защиты информации (РЗИ)**

**А.С.Красько**

**ОСНОВЫ КОМПЬЮТЕРНОГО  
ПРОЕКТИРОВАНИЯ И МОДЕЛИРОВАНИЯ  
РАДИОЭЛЕКТРОННЫХ СРЕДСТВ**

Учебное методическое пособие

**2012**

**Красько А.С.**

Основы компьютерного проектирования и моделирования радиоэлектронных средств: Учебное методическое пособие – Томск: ТУСУР, 2012. – 63 с.

Учебное методическое пособие содержит описание пяти лабораторных работ, выполняемых в ходе изучения дисциплины «Основы компьютерного проектирования и моделирования радиоэлектронных средств». Пособие содержит так же краткую вводную теоритическую часть, справочный материал по программам MatLab и Multisim, используемых при выполнении лабораторных работ, списки рекомендуемой литературы по разделам курса, листинги рабочих m-файлов. Пособие предназначено для студентов направления 210302-65 «Радиотехника» всех форм обучения.

© Красько А.С., 2012

## СОДЕРЖАНИЕ

Раздел	Заголовок	Стр.
1	Компьютерный лабораторный практикум	4
1.1	Задачи и содержание лабораторного практикума	4
1.2	Методические указания по выполнению лабораторных работ	4
2	Описание лабораторных работ	5
2.1	Краткие теоритические сведения	5
2.2	Лабораторная работа №1. Знакомство с системой MatLab	9
2.3	Лабораторная работа №2. Получение информации о схеме	10
2.4	Лабораторная работа №3. Формирование и решение уравнений схемы. Расчет АЧХ и ФЧХ	13
2.5	Лабораторная работа №4 Формирование и решение уравнений схемы. Расчет ПХ	20
2.6	Лабораторная работа №5. Расчет АЧХ, ФЧХ и ПХ усилительного каскада на биполярном и полевом транзисторах	23
3	Список рекомендуемой литературы	24
П1	Варианты заданий	26
П2	MatLab. Краткие сведения	28
П3	Multisim. Краткие сведения	48

# 1 КОМПЬЮТЕРНЫЙ ЛАБОРАТОРНЫЙ ПРАКТИКУМ

## 1.1 Задачи и содержание лабораторного практикума

Руководствуясь целями и задачами курса «ОКПиМРЭС» в целом, выделим основные задачи лабораторного практикума:

- 1) закрепить на практике реализацию основных алгоритмов анализа и расчета характеристик простых радиоэлектронных устройств, используя функциональную среду широко распространенной системы для инженерных и научных расчетов MatLab (не ниже версии 6.5);
- 2) ознакомиться на практике с моделями элементов и основными характеристиками простых радиоэлектронных устройств, как в частотной, так и во временной областях;
- 3) получить практические навыки автоматизированного проектирования простых устройств с заданными характеристиками, используя пакет схемотехнического проектирования Multisim не ниже 7-й версии.

## 1.2 Методические указания по выполнению лабораторных работ

Лабораторные работы выполняются в среде системы для инженерных и научных расчетов MatLab версии 6.5 и выше. Необходимые теоретические сведения, необходимые для выполнения лабораторных работ, содержатся в [1,2], в данном пособии приведен лишь минимально необходимый их объем. Сведения о системе MatLab и ее пакетах расширений можно почерпнуть из [3], либо любой другой по данной тематике, так же в Приложении П2 приведено краткое описание пакета.

Расчетные соотношения, необходимые для ручного расчета характеристик исследуемых цепей, приведены в описаниях соответствующих лабораторных работ настоящего пособия, там же даны ссылки на литературу.

Исследуемые схемы и варианты заданий на выполнение лабораторных работ находятся в приложении П1 данного пособия.

Для контроля результатов лабораторных работ можно воспользоваться известным пакетом для схемотехнического анализа и расчета Multisim версии 7.0 и выше[4], краткое описание которого приведено в Приложении П3.

Контроль результатов можно так же провести, используя элементарные соотношения из теории электрических цепей, часть которых приводятся в описаниях к лабораторным работам.

Отчет оформляется согласно правилам, принятым в настоящее время в ТУСУРе.

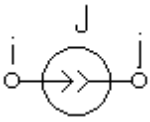
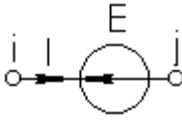
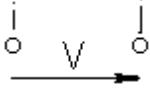
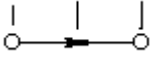
Отчет должен содержать:  
 формулировку задачи (по каждому пункту задания);  
 описание алгоритмов и программ;  
 листинги программ и распечатки результатов с достаточным для понимания количеством комментариев;  
 выводы по каждому пункту работы.

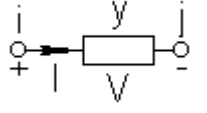
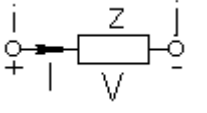
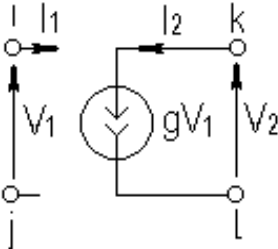
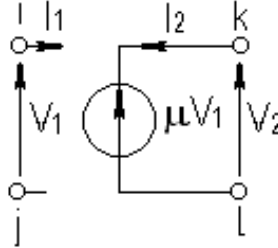
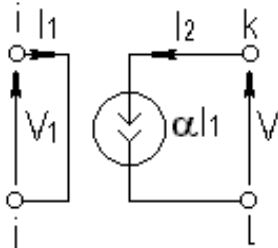
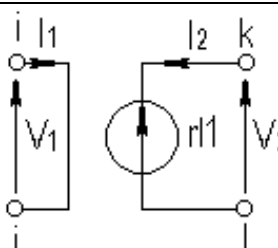
## 2. Описание лабораторных работ

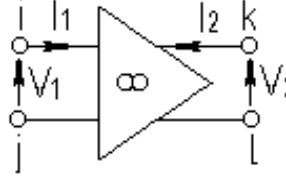
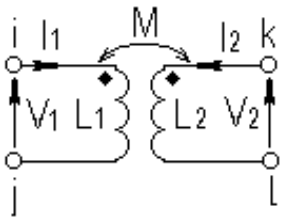
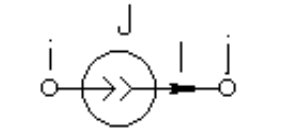
### 2.1 Краткие теоретические сведения

В лабораторных работах первой части используется модификация табличного метода формирования уравнений электронных цепей – модифицированный узловый метод с проверкой (МУПр). Метод достаточно подробно описан в [1]. Для удобства пользования данным пособием ограничимся кратким изложением теоретических основ метода и приведем таблицу представления идеальных элементов в модифицированном узловом методе с проверкой.

Таблица 2.1 – Представление идеальных элементов в модифицированном узловом методе с проверкой

Элем.	Обозначения	Матрица / вектор	Уравнения
Ист. тока 3-гр.		$i \begin{bmatrix} -J \\ \dots \\ j \end{bmatrix}$	$I_i = J;$ $I_j = -J.$
Ист. напр. 2-гр.		$v_i \quad v_j \quad I$ $i \begin{bmatrix} \cdot & \cdot & 1 \\ j \begin{bmatrix} \cdot & \cdot & -1 \\ m \begin{bmatrix} 1 & -1 & 0 \end{bmatrix} \end{bmatrix} \begin{bmatrix} \cdot \\ \cdot \\ E \end{bmatrix}$	$V_i - V_j = E;$ $I_i = I;$ $I_j = -I.$
Цепь XX		-----	$V = V_i - V_j.$
Цепь КЗ 2-гр.		$v_i \quad v_j \quad I$ $i \begin{bmatrix} \cdot & \cdot & 1 \\ j \begin{bmatrix} \cdot & \cdot & -1 \\ m \begin{bmatrix} 1 & -1 & 0 \end{bmatrix} \end{bmatrix}$	$V_i - V_j = 0;$ $I_i = I;$ $I_j = -I.$

Проводимость 1-гр.		$v_i \quad \cdot \quad v_j$ $i \begin{bmatrix} y & \cdot & -y \end{bmatrix}$ $\cdot \begin{bmatrix} \cdot & \cdot & \cdot \end{bmatrix}$ $j \begin{bmatrix} -y & \cdot & y \end{bmatrix}$	$I_i = y(V_i - V_j);$ $I_j = -y(V_i - V_j).$
Сопротивление 2-гр.		$v_i \quad v_j \quad I$ $i \begin{bmatrix} \cdot & \cdot & 1 \end{bmatrix}$ $j \begin{bmatrix} \cdot & \cdot & -1 \end{bmatrix}$ $m \begin{bmatrix} 1 & -1 & -z \end{bmatrix}$	$V_i - V_j - zI = 0;$ $I_i = I;$ $I_j = -I.$
ИТУН Ист. тока, упр. напр. 1-гр.		$v_i \quad \cdot \quad v_j$ $k \begin{bmatrix} g & \cdot & -g \end{bmatrix}$ $\cdot \begin{bmatrix} \cdot & \cdot & \cdot \end{bmatrix}$ $l \begin{bmatrix} -g & \cdot & g \end{bmatrix}$	$I_i = I_j = 0;$ $I_k = g(V_i - V_j);$ $I_l = -g(V_i - V_j).$
ИНУН Ист. напр., упр. напр. 2-гр.		$v_i \quad v_j \quad v_k \quad v_l \quad I$ $i \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & 0 \end{bmatrix}$ $j \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & 0 \end{bmatrix}$ $k \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & 1 \end{bmatrix}$ $l \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & -1 \end{bmatrix}$ $m \begin{bmatrix} -\mu & \mu & 0 & 0 & 0 \end{bmatrix}$	$I_k = -I_l = I_2;$ $\mu(V_i - V_j) =$ $v_k - v_l = V_2.$
ИТУТ Ист. тока, упр. током 2-гр.		$v_i \quad v_j \quad v_k \quad v_l \quad I$ $i \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & 1 \end{bmatrix}$ $j \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & -1 \end{bmatrix}$ $k \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \alpha \end{bmatrix}$ $l \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & -\alpha \end{bmatrix}$ $m \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \end{bmatrix}$	$I_i = -I_j = I_1;$ $I_k = -I_l = \alpha I_1;$ $v_i - v_j = 0.$
ИНУТ Ист. напр., упр. напр. 2-гр.		$v_i \quad v_j \quad v_k \quad v_l \quad I_1 \quad I_2$ $i \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & 1 & 0 \end{bmatrix}$ $j \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & -1 & 0 \end{bmatrix}$ $k \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & 0 & 1 \end{bmatrix}$ $l \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & 0 & -1 \end{bmatrix}$ $m \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 \end{bmatrix}$ $m+1 \begin{bmatrix} 0 & 0 & 1 & -1 & -r & 0 \end{bmatrix}$	$I_i = -I_j = I_1;$ $I_k = -I_l = I_2;$ $v_i - v_j = 0;$ $v_k - v_l - rI_1 = 0.$

<p>ОУ Опер. усил. 2-гр.</p>		$  \begin{matrix}  v_i & v_j & v_k & v_l & I \\  i & \begin{bmatrix} & & & & 0 \\ & & & & 0 \\ & & & & 1 \\ & & & & -1 \\ m & -\mu & \mu & 0 & 0 & 0 \end{bmatrix}  \end{matrix}  $	$  \begin{aligned}  I_k &= -I_l = I_2; \\  v_i - v_j &= 0.  \end{aligned}  $
<p>Транс- фор- матор 2-гр.</p>		$  \begin{matrix}  v_i & v_j & v_k & v_l & I_1 & I_2 \\  i & \begin{bmatrix} & & & & 1 & 0 \\ & & & & -1 & 0 \\ & & & & 0 & 1 \\ & & & & 0 & -1 \\ m & 1 & -1 & 0 & 0 & x_1 & x_m \\ m+1 & 0 & 0 & 1 & -1 & x_m & x_2 \end{bmatrix}  \end{matrix}  $	$  \begin{aligned}  I_i &= -I_j = I_1; \\  I_k &= -I_l = I_2; \\  v_i - v_j &= \\  sL_1 I_1 + sMI_2; \\  v_k - v_l &= \\  sMI_1 + sL_2 I_2.  \end{aligned}  $
<p>Ист. тока 2-гр.</p>		$  \begin{matrix}  v_i & v_j & I \\  i & \begin{bmatrix} \cdot & \cdot & 1 \\ \cdot & \cdot & -1 \\ m & \cdot & 0 \end{bmatrix} & \begin{bmatrix} \cdot \\ \cdot \\ J \end{bmatrix}  \end{matrix}  $	$  \begin{aligned}  I_i &= -I_j = I; \\  I &= J.  \end{aligned}  $

Идея модифицированного узлового метода с проверкой заключается в разделении элементов на группы; одна группа сформирована из элементов, которые описываются с помощью проводимостей, для элементов второй группы такое описание невозможно или нежелательно в силу ряда причин (например, при временном анализе индуктивность целесообразней представлять в виде сопротивления [1]). Поскольку через токи ветвей первой группы можно выразить напряжения ветвей, а напряжения ветвей - через узловые потенциалы, то можно исключить из табличных уравнений все напряжения ветвей, а для элементов первой группы еще и токи ветвей. При введении дополнительных уравнений для токов в ветвях с элементами второй группы производится проверка на наличие заранее известных (нулевых) переменных. В результате такого преобразования получим уравнения модифицированного узлового метода с проверкой

$$\begin{matrix}
 n & m \\
 \begin{bmatrix} Y_{n1} & y_{p-} \\ \text{доп.} & \text{-ния} \end{bmatrix} \cdot \begin{bmatrix} V_n \\ I_2 \end{bmatrix} = \begin{bmatrix} J_n \\ W_2 \end{bmatrix}
 \end{matrix}$$

или в общем виде

$$T \cdot X = W,$$



где  $n$  - размерность матрицы проводимости  $Y_{n1}$  элементов первой группы ( $n$  - число узлов схемы без нулевого);  $m$  - число дополнительных уравнений для элементов второй группы;  $J_n$  - вектор независимых источников тока;  $I_2$  - вектор токов ветвей элементов второй группы;  $W_2$  - вектор, куда входят независимые источники напряжения, а также начальные напряжения и токи на конденсаторах и индуктивностях (индуктивности представлены элементами второй группы).

При частотном анализе электронных схем организуется цикл по частоте, внутри которого для каждой частотной точки формируется система уравнений, которая решается относительно интересующих напряжений и токов.

При временном анализе линейных электронных схем возможно непосредственно использовать модифицированную узловую форму уравнений

$$(G + pC) \cdot X = W.$$

После перехода во временную область получим

$$G \cdot x + C \cdot x' = W,$$

или

$$C \cdot x' = W - G \cdot x.$$

Решение полученной системы дифференциальных уравнений находится путем численного интегрирования. Одними из эффективных методов численного интегрирования являются методы, опирающиеся на **линейные многошаговые формулы** [1], к простейшим из которых относятся формулы Эйлера (прямая и обратная) и формула трапеций.

Разбив временной интервал  $(0, T)$  на конечное число отрезков  $h$  и положив  $t_{n+1} = t_n + h$ , для каждого момента времени  $t_n$  можно найти приближение  $x_n$  к истинному решению  $x(t_n)$  путем применения линейных многошаговых формул:

$$x_{n+1} = x_n + hx_n' \quad (\text{прямая формула Эйлера});$$

$$x_{n+1} = x_n + hx_{n+1}' \quad (\text{обратная формула Эйлера});$$

$$x_{n+1} = x_n + (h/2)(x_n' + x_{n+1}') \quad (\text{формула трапеций}).$$

Нахождение  $x_{n+1}'$  для  $(n+1)$ -го шага вычислений возможно путем применения прямой формулы Эйлера.

Поскольку напряжение на конденсаторе и ток, протекающий через него, связаны соотношением  $i = C \cdot dv/dt$ , а для индуктивности имеем  $v = L \cdot di/dt$ , то применение обратной формулы Эйлера равноценно переходу от емкостей и индуктивностей к их эквивалентным схемам, показанным на рисунке 5.1, в результате чего цепь становится резистивной.

Такие модели индуктивности и емкости носят название **сеточных (сопровождающих, дискретных) моделей**.

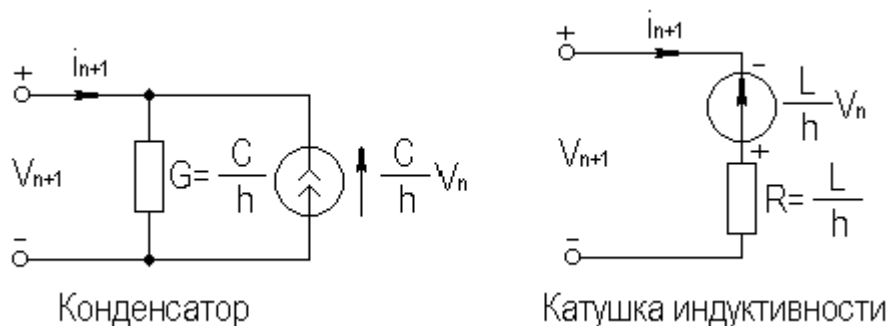


Рисунок 2.1 – Сеточные модели

Следует заметить, что применение сеточных моделей не приводит к изменению размерности системы схемных уравнений, поскольку значения токов и напряжений эквивалентных генераторов заносятся в соответствующие ячейки вектора свободных членов. Получаемая при этом матрица коэффициентов  $T$  носит вещественный характер.

При временном анализе электронных схем организуется цикл по времени, внутри которого для каждой временной точки формируется система уравнений, которая решается относительно интересующих напряжений и токов. Для первого шага значения напряжений и токов эквивалентных генераторов сеточных моделей принимаются равными начальным, чаще всего, нулевым.

## 2.2 Лабораторная работа №1. Знакомство с системой MATLAB

**Цель работы.** Знакомство с некоторыми возможностями и получение первоначальных навыков программирования в среде системы MATLAB .

**Необходимые теоретические сведения.** Для первоначального знакомства с системой MATLAB достаточно обратиться к [3], или к другой подобной литературе. Краткие сведения по программированию в системе MATLAB излагаются в последующих лабораторных работах.

### Задание на работу:

1. Создать в системе MATLAB несколько массивов (в том числе и со смешанными типами элементов (например, текстовый и числовой)). Для версии 6.5 и выше показать возможности класса ячеек [3].

2. Сформировать матрицу коэффициентов и вектор свободных членов системы линейных алгебраических уравнений (любых, по выбору студента,

но не ниже третьего порядка) и решить эту систему представленными в пакете MATLAB способами. Произвести проверку правильности решения.

3. Получить графики нескольких функции (любых, по усмотрению студента). Показать возможность построения двух графиков в одном графическом окне («фигуре»).

4. Оформить программы, выполненные в пунктах 1, 2, 3 в виде *файл – функций (процедур)*. Написать и отладить *script – файл (файл – сценарий, или управляющую программу)*, в котором с помощью функции *теги* реализовать возможность выбора любой из программ 1, 2, 3 и выхода из программы.

**Содержание отчета.** Отчет оформляется согласно общим правилам. Программы и распечатки результатов должны быть снабжены достаточным для понимания количеством комментариев. В выводах отметить схожесть и отличие языка программирования системы MATLAB от изучаемых ранее в курсе «Информатика» языков программирования.

### 2.3 Лабораторная работа №2. Получение информации о схеме

**Цель работы.** Целью работы является знакомство со способами задания информации о топологии схемы и работа с программой, позволяющей автоматически определять топологические характеристики электронной схемы.

**Необходимые теоретические сведения.** Для формирования уравнений модифицированным узловым методом с проверкой необходима информация о количестве узлов схемы и о количестве ветвей второго рода (см. подраздел 2.1). Эта информация необходима для определения порядка системы схемных уравнений. Желательно, чтобы эта информация определялась автоматически каждый раз, как только в программу вводились данные об элементах исследуемой схемы.

Примем следующую кодировку двухполюсных элементов схемы:

**признак, номинал, узлы подключения**, т.е. двухполюсный элемент будет описываться четырьмя элементами (ветвь КЗ - тремя, ввиду отсутствия позиции номинала).

В качестве признака будем использовать строчные или прописные буквы латинского алфавита.

Условимся о следующей кодировке признака элемента:

Емкость как проводимость -	<b>C</b>
Проводимость -	<b>G</b>
Сопротивление -	<b>R</b>
Короткозамкнутая ветвь -	<b>K</b>

Индуктивность как сопротивление -	<b>L</b>
Источник тока -	<b>I</b>
Источник ЭДС -	<b>E</b>
Источник тока, управляемый напряжением (ИТУН) -	<b>S</b>
Источник тока, управляемый током (ИТУТ) -	<b>A</b>
Идеальный ОУ -	<b>O</b>

В этом случае информация, например, об конденсаторе С схемы рисунка 5.2, представится следующим образом (через пробелы):

**'C' 25e-9 2 0**

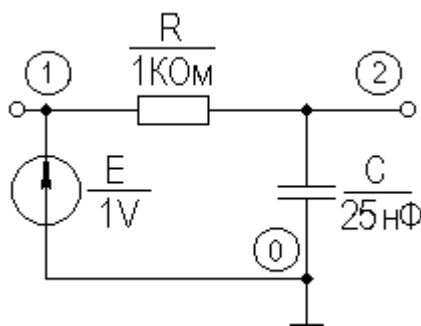


Рисунок 2.2 – Интегрирующая цепь

Условимся **всегда** обозначать общий узел как **нулевой**. Номиналы элементов представляются в основных единицах:

R - Ом, C - Ф, L - Генри, Y – См.

Для схемы рисунка 5.2 файл данных **dd** будет выглядеть следующим образом:

`dd={'E' 1 1 0 'R' 1e3 1 2 'C' 25e-9 2 0};`

(вариант для версии MatLab 5.0 и выше, для низших версий фигурные скобки заменяются на квадратные).

Четырехполюсные элементы, например ИТУН (рисунок 5.3), кодируются так:

**признак, номинал, узлы управления, выходные узлы,** и содержат шесть элементов (исключение составляет идеальный ОУ, информация о нем содержит пять элементов ввиду отсутствия позиции «номинал»).

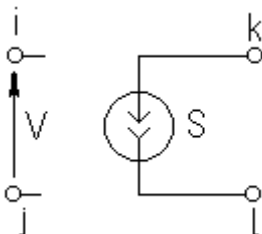


Рисунок 2.3 – ИТУН

Предположим, что  $S=10$  мСм;  $i=3$ ;  $j=0$ ;  $k=4$ ;  $l=5$ . В таком случае информация о ИТУН представится в следующем виде (через пробелы):

**'S' 1e-2 3 0 4 5**

Рассмотренная ниже файл-функция **top**, на основании информации об элементах схемы (массив **dd**), определяет максимальное количество узлов схемы **n** и число ветвей второй группы элементов **v2**

*Файл - функция определения порядка схемных уравнений для МУПр*

```
function [n,v2]=top(dd);
% определение числа узлов n и числа ветвей второго рода v2
% dd-входной массив-вектор данных об элементах схемы
% kp-шаг цикла; kp, bn-вспомогательные переменные
nm=max(size(dd)); % определение размерности массива dd
k=1; n=0; v2=0;
while k<=nm; % перебор элементов схемы с шагом kp
    pr=upper(dd{k}); % выделение признака элемента схемы
    if (pr=='R')|(pr=='L')|(pr=='E'); % элементы R, L, E
        kd=k+3; kp=4; v2=v2+1;
    end;
    if (pr=='C')|(pr=='G')|(pr=='I'); % элементы C, G, I
        kd=k+3; kp=4;
    end;
    if (pr=='S'); % ИТУН
        kd=k+5; kp=6;
    end;
    if (pr=='A'); % ИТУТ
        kd=k+5; kp=6; v2=v2+1;
    end;
    bn=k+2;
    if (pr=='K'); % ветвь КЗ
        kd=k+2; kp=3; v2=v2+1; bn=k+1;
    end;
    if (pr=='O'); % идеальный ОУ
        kd=k+4; kp=5; v2=v2+1; bn=k+1; end;
    for k4=bn : kd; % определение n путем выбора максимального номера узла
        if ceil(dd{k4}) > n;
            n=ceil(dd{k4});
        end;
    end;
    k=k+kp;
end;
end;
```

**Задание на работу:**

1. Составить script – файл с файл – функцией **top**, в котором предусмотреть:

ввод исходных данных (файл **dd** допускается задавать путем присвоения значений непосредственно в теле script – файла, организация загрузки исходных данных из отдельного файла относится к «высшему пилотажу» и всячески приветствуется);

выбор с помощью функции **menu** исследуемой схемы из набора схем Приложения В с данными вашего варианта работы;

вывод количества узлов и количества ветвей второго рода.

2. Модернизировать файл – функцию **top** с целью получения дополнительной информации об общем количестве элементов схемы и количестве элементов первой группы.

3. Ввести данные об элементах и провести анализ топологии **всех** схем Приложения А.

**Содержание отчета.** Отчет оформляется согласно общим требованиям. Программы и распечатки результатов должны быть снабжены достаточным для понимания количеством комментариев. В выводах отметить внесенные изменения и корректность работы программы.

## 2.4 Лабораторная работа №3. Формирование и решение уравнений схемы. Расчет АЧХ и ФЧХ

**Цель работы.** Составление программы анализа электронных схем в частотной области с использованием разработанной в предыдущей работе подпрограммы топологического анализа электронных цепей и подпрограммы формирования схемных уравнений, рассматриваемой в данной работе. Отладка и тестирование созданной программы, расчет АЧХ и ФЧХ простейших цепей.

**Необходимые теоретические сведения.** В разделе 2.1 дано краткое описание одного из методов формирования схемных уравнений – МУПр и приведена таблица 2.1, следуя которой, идеальные схемные элементы заносятся в матрицу коэффициентов схемных уравнений данного метода.

Рассмотренная ниже файл – функция **matr** на основании массива – вектора исходных данных об элементах схемы **dd** и определенных с помощью файл – функции **top** данных о количестве узлов схемы **n** и ветвей второй группы **v2**, формирует матрицу коэффициентов **t** и вектор свободных членов **ww** для фиксированной частотной точки **w**.

Алгоритм формирования достаточно прозрачен и состоит из следующих операций:

- 1) предварительного обнуления матрицы **t** и вектора **ww**;
- 2) организации цикла по элементам с определением их типа, выделением номинала и узлов подключения;

3) занесения информации об элементе в матрицу **t** (по необходимости, и в вектор **ww**), согласно правилам таблицы 2.1.

Следует помнить, что MATLAB не позволяет работать с нулевыми индексами массивов, обойти это ограничение возможно путем задания в программе дополнительных условий на равенство нулю номеров узлов подключения элементов.

Нелишне напомнить, что вычисление индуктивного сопротивления и емкостной проводимости проводится по следующим формулам:

$$Z_L = j\omega L; \quad Y_C = j\omega C.$$

где  $\omega = 2\pi f$ .

**Файл – функция формирования матрицы коэффициентов и вектора свободных членов для МУПр.**

```
function[t,ww]=matr(dd,n,v2,w);
% функция формирования матрицы коэффициентов t и вектора свободных
% членов ww
mn=max(size(dd)); % определение макс. размера массива данных dd
t=zeros(n+v2,n+v2); % обнуление матрицы коэффициентов t и вектора
ww=zeros(n+v2,1); % свободных членов ww
k=1; du=1; % du – количество дополнительных уравнений
while k<=mn; % цикл по перебору элементов
    pr=upper(dd{k}); % выделение признака элемента
    kp=4; % количество позиций в описании элемента
    if (pr=='O') | (pr=='K');
        n1=ceil(dd{k+1}); n2=ceil(dd{k+2}); % выделение узлов подключения
    end;
    if (pr=='R') | (pr=='G') | (pr=='C') | (pr=='L') |...
        (pr=='E') | (pr=='I') | (pr=='S') | (pr=='A');
        n1=ceil(dd{k+2}); n2=ceil(dd{k+3}); zn=abs(dd{k+1});
    end;
    % выделение узлов подключения и номинала
    % формирование t и ww
    % Емкость как проводимость - C
    if pr=='C';
        if n1~=0;
            t(n1,n1)=t(n1,n1)+j*zn*w;
        end;
        if n2~=0;
            t(n2,n2)=t(n2,n2)+j*zn*w;
        end;
        if (n1~=0)&(n2~=0);
            t(n1,n2)=t(n1,n2)-j*zn*w; t(n2,n1)=t(n2,n1)-j*zn*w;
```

```

    end;
end;
% Проводимость - G
if pr=='G';
    if n1~=0;
        t(n1,n1)=t(n1,n1)+zn;
    end;
    if n2~=0;
        t(n2,n2)=t(n2,n2)+zn;
    end;
    if (n1~=0)&(n2~=0);
        t(n1,n2)=t(n1,n2)-zn; t(n2,n1)=t(n2,n1)-zn;
    end;
end;
% Сопротивление - R
if pr=='R';
    if n1~=0;
        t(n+du,n1)=t(n+du,n1)+1; t(n1,n+du)=t(n1,n+du)+1;
    end;
    if n2~=0;
        t(n+du,n2)=t(n+du,n2)-1; t(n2,n+du)=t(n2,n+du)-1;
    end;
    t(n+du,n+du)=t(n+du,n+du)-zn; du=du+1;
end;
% Индуктивность как сопротивление - L
if pr=='L';
    if n1~=0;
        t(n+du,n1)=t(n+du,n1)+1; t(n1,n+du)=t(n1,n+du)+1;
    end;
    if n2~=0;
        t(n+du,n2)=t(n+du,n2)-1; t(n2,n+du)=t(n2,n+du)-1;
    end;
    t(n+du,n+du)=t(n+du,n+du)-j*w*zn; du=du+1;
end;
% Источник ЭДС - E
if pr=='E';
    if n1~=0;
        t(n+du,n1)=t(n+du,n1)+1; t(n1,n+du)=t(n1,n+du)+1;
    end;
    if n2~=0;
        t(n+du,n2)=t(n+du,n2)-1; t(n2,n+du)=t(n2,n+du)-1;
    end;
    ww(n+du,1)=ww(n+du,1)+zn; du=du+1;

```



```

end;
% Источник тока - I
if pr=='I';
    if n1~=0;
        ww(n1,1)=ww(n1,1)+zn;
    end;
    if n2~=0;
        ww(n2,1)=ww(n2,1)-zn;
    end;
end;
% Ветвь K3 - K
if pr=='K';
    if n1~=0;
        t(n+du,n1)=t(n+du,n1)+1; t(n1,n+du)=t(n1,n+du)+1;
    end;
    if n2~=0;
        t(n+du,n2)=t(n+du,n2)-1; t(n2,n+du)=t(n2,n+du)-1;
    end;
    du=du+1; kp=3;
end;
% Упр. источники
if (pr=='S') | (pr=='A');
    n3=ceil(dd{k+4});n4=ceil(dd{k+5});
    % ИТУН - S
    if pr=='S';
        if n3~=0;
            if n1~=0;
                t(n3,n1)=t(n3,n1)+zn;
            end;
            if n2~=0;
                t(n3,n2)=t(n3,n2)-zn;
            end;
        end;
        if n4~=0;
            if n2~=0;
                t(n4,n2)=t(n4,n2)+zn;
            end;
            if n1~=0;
                t(n4,n1)=t(n4,n1)-zn;
            end;
        end;
    end;
    du=du+1; kp=6;
end;

```

```

% ИТУТ - А
if pr=='A';
    zn=abs(dd{k+1});
    if n1~=0;
        t(n+du,n1)=t(n+du,n1)+1; t(n1,n+du)=t(n1,n+du)+1;
    end;
    if n2~=0;
        t(n+du,n2)=t(n+du,n2)-1; t(n2,n+du)=t(n2,n+du)-1;
    end;
    if n3~=0;
        t(n3,n+du)=t(n3,n+du)+zn;
    end;
    if n4~=0;
        t(n4,n+du)=t(n4,n+du)-zn;
    end;
    du=du+1; kp=6;
end;
end;
% ОУ - О
if pr=='O';
    n3=ceil(dd{k+3}); n4=ceil(dd{k+4});
    if n1~=0;
        t(n+du,n1)=t(n+du,n1)+1;
    end;
    if n2~=0;
        t(n+du,n2)=t(n+du,n2)-1;
    end;
    if n3~=0;
        t(n3,n+du)=t(n3,n+du)+1;
    end;
    if n4~=0;
        t(n4,n+du)=t(n4,n+du)-1;
    end;
    du=du+1; kp=5;
end; % Переход к следующему элементу
k=k+kp;
end;

```

Результатом решения схемных уравнений МУПр являются узловые потенциалы и токи ветвей второй группы.

Коэффициент передачи схемы определяется как отношение напряжения на выходном узле к напряжению на входном (естественно, эти узлы необходимо задать в главной программе). Следует заметить, что данное

определение справедливо при наличии общего для входа и выхода узла (в нашем случае нулевого).

Коэффициент передачи для цепей с реактивностями имеет комплексный характер:

$$\dot{K} = U_{\hat{a}\hat{b}\hat{o}} / U_{\hat{a}\hat{o}} = |K| \exp(j\varphi),$$

где  $\varphi$  - фазовый сдвиг между входным и выходным сигналами.

Зависимость  $|K|$  от частоты называется амплитудно-частотной характеристикой (АЧХ), а  $\varphi$  от частоты – фазо-частотной (ФЧХ). Строятся АЧХ и ФЧХ в полулогарифмическом масштабе (по оси частот).

Расчет АЧХ и ФЧХ можно проводить по следующему алгоритму:

- 1) вводится информация об элементах схемы и номерах входного и выходного узлов схемы;
- 2) определяются количество узлов и ветвей второй группы;
- 3) организуется цикл по частоте с заданным шагом;
- 4) внутри цикла для каждой точки частотного диапазона формируется матрица коэффициентов и вектор свободных членов системы схемных уравнений, решается эта система, вычисляется коэффициент передачи, определяется модуль и фаза коэффициента передачи;
- 5) накапливаются массивы значений модуля, фазы, точек частотного диапазона и строятся АЧХ и ФЧХ.

Программа тестируется путем расчета АЧХ и ФЧХ простейших электрических цепей, схемы которых приведены на рисунках 2.4, 2.5, данные о номиналах элементов схемы берутся из соответствующей таблицы вариантов в приложении А.

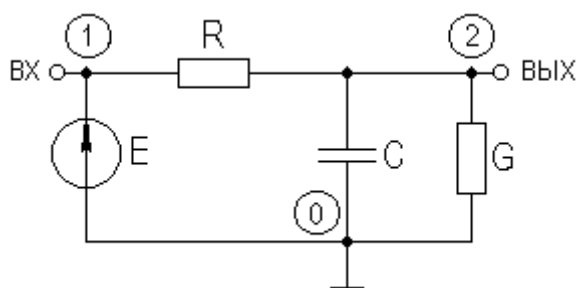


Рисунок 2.4 – Схема делителя с емкостью

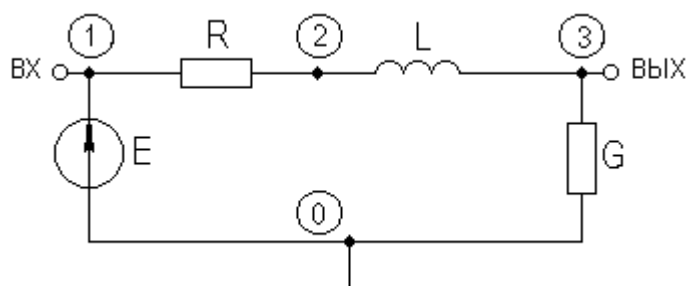


Рисунок 2.5 – Схема делителя с индуктивностью

Рассмотрим в качестве примера расчет АЧХ и ФЧХ интегрирующей цепи (рисунок 2.2). Очевидно, что модуль коэффициента передачи на нулевой частоте для этой схемы  $K_0 = 1$ , с ростом частоты его значение уменьшается. Значение верхней граничной частоты по уровню 0,707 определится как:

$$f_a = 1/2\pi\tau,$$

где  $\tau$  - постоянная времени цепи,  $\tau = RC$ . Подставив в выражение для  $f_a$  значения элементов, указанные на рисунке 2.2, получим значение  $f_a \approx 6,4 \cdot 10^3 \text{ Гц}$ .

Результаты расчета интегрирующей цепи программой, составленной по данным рекомендациям и с использованием приведенных процедур, приведены на рисунке 2.6.

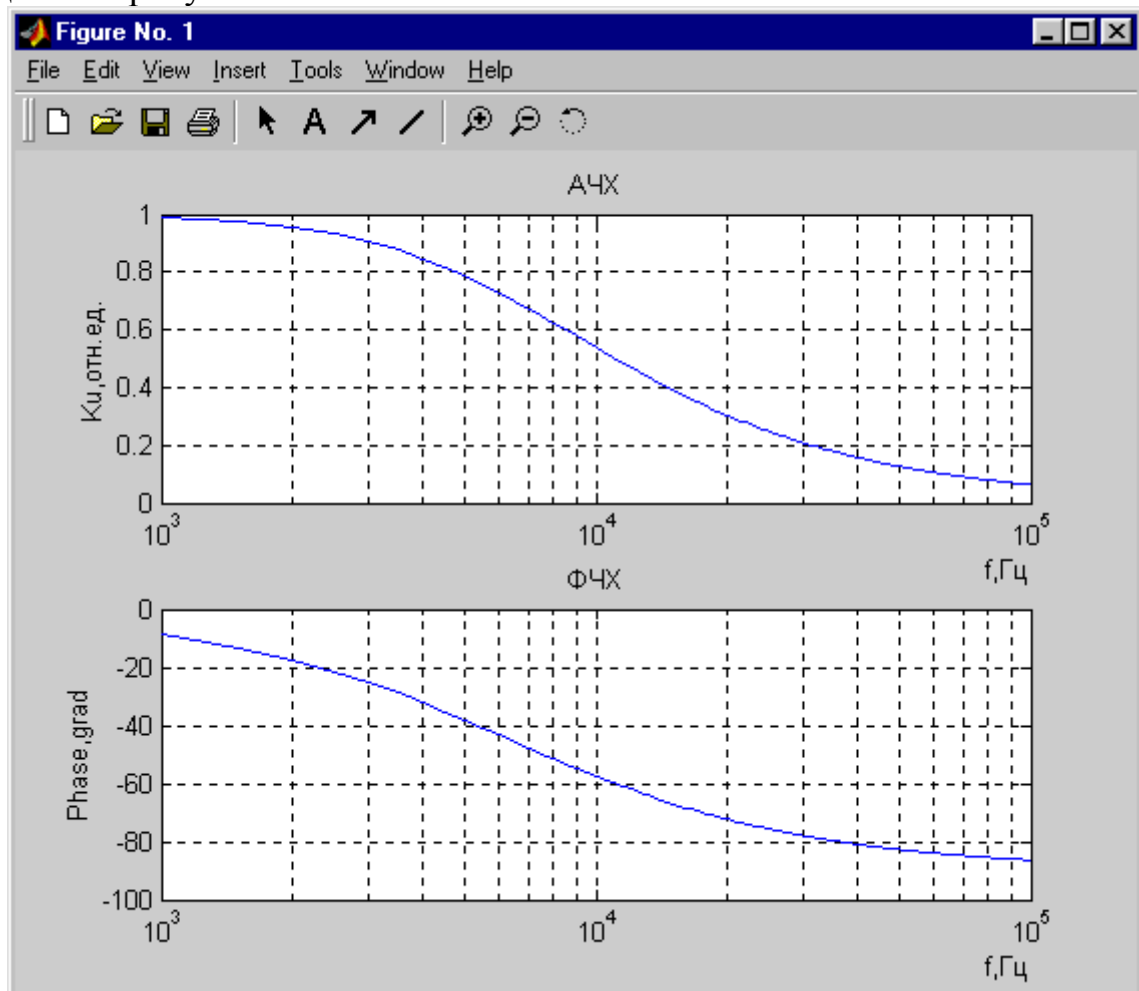


Рисунок 2.6 – АЧХ и ФЧХ интегрирующей цепи.

Как видно из результатов расчета, значение  $f_a$  соответствует расчетному, характер и величина фазового сдвига не противоречит теории.

Частотный диапазон следует выбирать исходя из ожидаемого  $f_g$ , беря нижнюю и верхнюю его границы, соответственно, примерно на порядок ниже и выше  $f_g$ . Шаг по частоте рекомендуется брать таким, чтобы в пределах частотного диапазона было не менее 20 точек отсчета.

#### **Задание на работу:**

1. Составить script – файл для расчета АЧХ и ФЧХ цепей с файл – функциями **top** и **matr**. Использовать функцию **menu** для выбора схемы и графический вывод АЧХ и ФЧХ.

2. Провести расчет АЧХ и ФЧХ схем рисунков 2.4 и 2.5 с данными вашего варианта.

3. Ввести в любую из рассчитанных схем короткозамкнутую ветвь, провести расчет АЧХ и ФЧХ этой схемы.

3. С помощью пакета Multisim доказать корректность работы программы.

**Содержание отчета.** Отчет оформляется согласно общим требованиям. Программы и распечатки результатов должны быть снабжены достаточным для понимания количеством комментариев. В выводах привести результаты расчетов АЧХ и ФЧХ исследуемых схем и доказательство корректности работы программы.

## **2.5 Лабораторная работа №4. Формирование и решение уравнений схемы. Расчет ПХ**

**Цель работы.** Составление программы анализа электронных схем в временной области на основе разработанной в предыдущих работах подпрограмм топологического анализа электронных цепей и формирования схемных уравнений. Отладка и тестирование созданной программы, расчет ПХ простейших цепей, для которых были рассчитаны АЧХ и ФЧХ.

**Необходимые теоретические сведения.** В подразделе 2.1 дано краткое описание метода временного анализа электронных цепей с использованием сеточных моделей.

Ниже приводится фрагмент файл - функции, предназначенной для формирования схемных уравнений во временной области.

Фрагмент файл – функции **matrt** для записи в матрицу коэффициентов **t** и вектора свободных членов **ww** данных об реактивных элементах схемы:

%Емкость как проводимость - C

if pr=='C';

zn=zn/dt; % dt – шаг по времени, передается из основной программы в matrt

if n1~=0;

t(n1,n1)=t(n1,n1)+zn;

```

ww(n1)=ww(n1)+uv(n1)*zn; % uv – массив узловых напряжений,
%обнуляется в основной программе и передается в matr
end;
if n2~=0;
t(n2,n2)=t(n2,n2)+zn;
ww(n2)=ww(n2)+uv(n2)*zn;
end;
if (n1~=0) & (n2~=0);
t(n1,n2)=t(n1,n2)-zn;t(n2,n1)=t(n2,n1)-zn;
ww(n1)=ww(n1)-zn*(uv(n1)-uv(n2));
ww(n2)=ww(n2)-zn*(uv(n1)-uv(n2));
end;
end;
% Индуктивность как сопротивление - L
if pr=='L';
zn=zn/dt;
if n1~=0;
t(n+du,n1)=t(n+du,n1)+1; t(n1,n+du)=t(n1,n+du)+1;
end;
if n2~=0;
t(n+du,n2)=t(n+du,n2)-1; t(n2,n+du)=t(n2,n+du)-1;
end;
t(n+du,n+du)= t(n+du,n+du)-zn;
ww(n+du)= ww(n+du)-zn*uv(n+du);
du=du+1;
end;

```

В остальном функция **matrt** ничем не отличается от функции **matr**. Переходная характеристика характеризует зависимость изменения выходного напряжения цепи при подаче на вход единичного скачка напряжения и оценивается путем определения времени установления  $t_{\sigma}$ . Время установления  $t_y$  - это время, за которое выходное напряжение нарастает от уровня 0,1 до уровня 0,9 от установившегося значения.

Расчет ПХ можно проводить по следующему алгоритму:

- 1) вводится информация об элементах схемы и номера входных и выходных узлов;
- 2) определяются количество узлов и ветвей второй группы;
- 3) обнуляется вектор узловых потенциалов **ww** и массив узловых напряжений **uv**;
- 4) организуется цикл по времени с заданным шагом **dt**;
- 5) внутри цикла для каждой точки временного диапазона формируется матрица коэффициентов и вектор свободных членов системы схемных

уравнений, решается эта система, вычисляется значение выходного напряжения;

б) накапливается массив значений выходного напряжения, точек временного диапазона и строится ПХ.

Рассмотрим в качестве примера расчет ПХ интегрирующей цепи (рисунок 2.2). Очевидно, что выходное напряжение для этой схемы стремится к единице. Значение времени установления определится как:

$$t_{\delta} = 2,2 \cdot \tau,$$

где  $\tau$  - постоянная времени цепи,  $\tau = RC$ . Подставив в выражение для  $t_{\delta}$  значения элементов, указанные на рисунке 2.2, получим значение  $t_{\delta} \approx 0,55 \cdot 10^{-4} \text{ н}$ .

Результаты расчета интегрирующей цепи программой, составленной по данным рекомендациям и с использованием приведенных процедур, приведены на рисунке 2.7.

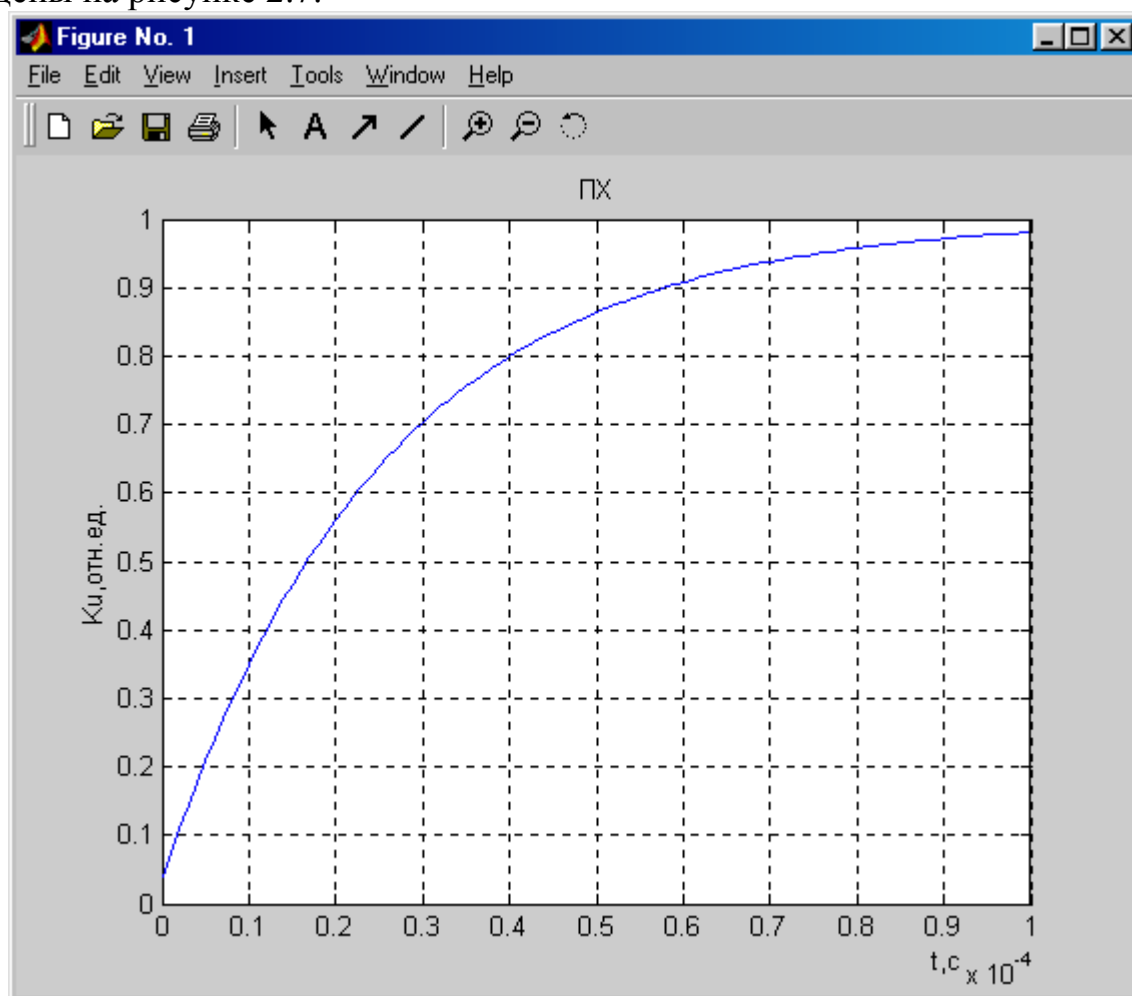


Рисунок 2.7 –ПХ интегрирующей цепи.

Как видно из результатов расчета, значение  $t_y$  соответствует расчетному. Временной диапазон следует выбирать исходя из ожидаемого  $t_y$ , задавая его в пределах от нуля, до  $(2 \div 3) \cdot t_y$ . Шаг по времени  $dt$  рекомендуется брать таким, чтобы в пределах временного диапазона было не менее 20 точек отсчета.

#### **Задание на работу:**

1. Составить script – файл для расчета ПХ цепи с файл – функциями **top** и **matrt**. Использовать функцию **menu** для выбора схемы и графический вывод ПХ.

2. Провести расчет ПХ схем рисунков 2.4 и 2.5 с данными вашего варианта.

4. Ввести в любую из рассчитанных схем короткозамкнутую ветвь, провести расчет ПХ этой схемы.

3. С помощью пакета Multisim доказать корректность работы программы.

**Содержание отчета.** Отчет оформляется согласно общим требованиям. Программы и распечатки результатов должны быть снабжены достаточным для понимания количеством комментариев. В выводах привести результаты расчетов ПХ исследуемых схем и доказательство корректности работы программы.

## **2.6 Лабораторная работа №5. Расчет АЧХ, ФЧХ и ПХ усилительных каскадов на биполярном и полевом транзисторах**

**Цель работы.** Анализа электронных схем в частотной и временной областях на основе разработанных в предыдущих работах программ. Расчет АЧХ, ФЧХ и ПХ электронных цепей, содержащих идеальные полевой (ПТ) и биполярный (БТ) транзисторы, моделируемые, соответственно, ИТУН и ИТУТ.

**Необходимые теоретические сведения.** Каскады на идеальных ПТ и БТ, моделируемые ИТУН и ИТУТ соответственно, представлены на рисунках 5.8 и 5.9.

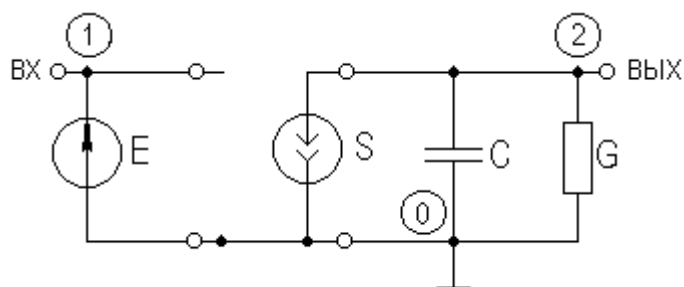




Рисунок 2.8 – Каскад на ПТ

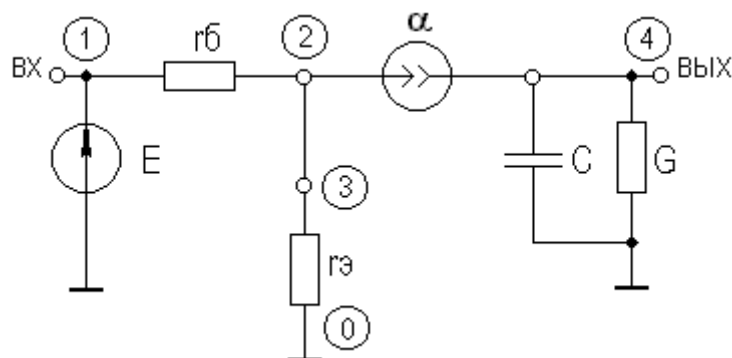


Рисунок 2.9 – Каскад на БТ

Модуль коэффициента передачи на нулевой частоте для схемы рисунка 5.8  $\hat{E}_0 = S/G$ , с ростом частоты его значение уменьшается. Значение верхней граничной частоты по уровню 0,707 определится как:

$$f_a = 1/2\pi\tau,$$

где  $\tau$  - постоянная времени цепи,  $\tau = C/G$ .

Значение фазового сдвига на нулевой частоте  $\varphi_0 = 180^0$ , на  $f_a$  фазовый сдвиг  $\varphi_0 \approx 135^0$  и далее с ростом частоты он убывает.

Аналогичным образом рассчитывается каскад на БТ, если вначале определить значение крутизны как:

$$S = \frac{\alpha}{r_y + (1 - \alpha) \cdot r_a}.$$

Очевидно, что выходное напряжение для этих схем стремится к  $U_{\hat{a}\hat{o}} = K_0 \cdot U_{\hat{a}\hat{o}}$ . Значение времени установления определится как:

$$t_{\hat{o}} = 2,2 \cdot \tau = 0,35 / f_a,$$

где  $\tau$  - постоянная времени цепи,  $\tau = C/G$ .

#### Задание на работу:

1. Доработать программу с целью организации с помощью функции **menu** возможности выбора:

а) рассчитываемой схемы; б) АЧХ и ФЧХ, либо ПХ.

2. Провести расчет АЧХ и ФЧХ и ПХ схем рисунков 5.4 и 5.5 с данными вашего варианта.

3. По выбору студента рассчитать схему с элементом, который не был представлен в рассчитанных ранее схемах.

3. С помощью пакета Multisim доказать корректность работы программы.

**Содержание отчета.** Отчет оформляется согласно общим требованиям. Программы и распечатки результатов должны быть снабжены достаточным для понимания количеством комментариев. В выводах привести результаты расчетов ПХ исследуемых схем и доказательство корректности работы программы.

### **3 СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ**

1. Кологривов В.А. Основы автоматизированного проектирования радиоэлектронных устройств: Учебное пособие для студентов специальности 200700.- Томск: ТУСУР, 2002. - 247с.: ил.
2. Влах И., Сингхал К. Машинные методы анализа и проектирования электронных схем. - М.: Радио и связь, 1988. - 560 с.
3. Дьяконов В.П. MatLab 7\*/R2006/R207. Самоучитель. М.:ДМК-пресс, 2008 – 768с.:ил.
4. Хернитер Марк Е. Multisim<sup>®</sup> современная система компьютерного моделирования и анализа схем электронных устройств. – Москва, ДМК-пресс, 2006. – 488с.:ил.

## Приложение А

### Варианты заданий

На рисунке А1 приведены схемы электрических цепей, исследуемых в лабораторных работах 1÷6.

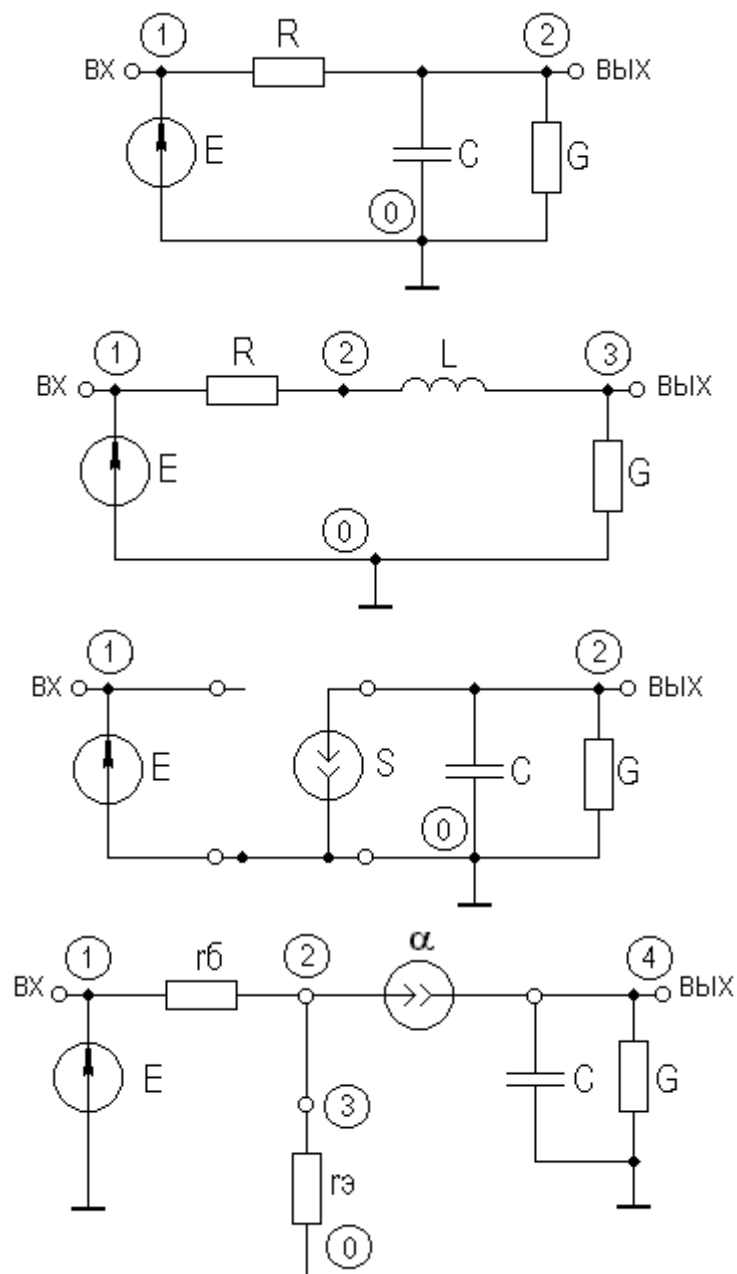


Рисунок А1 – Схемы, исследуемые в работах 1÷6

Данные номиналов элементов, **общие** для всех работ, приведены в таблице А1.

Таблица А1 Общие номиналы элементов схем

элемент	E	R	S	$r_{\dot{a}}$	$r_{\dot{y}}$	$\alpha$
номинал	1 В	1 кОм	100мСм	20 Ом	10 Ом	0,99

Данные номиналов элементов, зависящие от **варианта** задания, приведены в таблице А2.

Таблица А2 Варианты номиналов исследуемых схем

элемент вариант	G мСм	C нФ	L нГн
1	20	1	1
2	19	2	2
3	18	3	3
4	17	4	4
5	16	5	5
6	15	6	6
7	14	7	7
8	13	8	8
9	12	9	9
10	11	10	10
11	10	11	11
12	9	12	12
13	8	13	13
14	7	14	14
15	6	15	15
16	5	16	16
17	4	17	17
18	3	18	18
19	2	19	19
20	1	20	20

## Приложение Б

### MatLab

#### Краткие сведения

MATLAB - это высокопроизводительный язык для технических расчетов. Он включает в себя вычисления, визуализацию и программирование в удобной среде, где задачи и решения выражаются в форме, близкой к математической. Типичное использование MATLAB - это:

математические вычисления

создание алгоритмов

моделирование

анализ данных, исследования и визуализация

научная и инженерная графика

разработка приложений, включая создание графического интерфейса

MATLAB - это интерактивная система, в которой основным элементом данных является массив. Это позволяет решать различные задачи, связанные с техническими вычислениями, особенно в которых используются матрицы и вектора, в несколько раз быстрее, чем при написании программ с использованием "скалярных" языков программирования, таких как Си или Фортран.

MATLAB развивался в течении нескольких лет, ориентируясь на различных пользователей. В университетской среде, он представлял собой стандартный инструмент для работы в различных областях математики, машиностроения и науки. В промышленности, MATLAB - это инструмент для высокопродуктивных исследований, разработок и анализа данных.

В MATLAB важная роль отводится специализированным группам программ, называемых *toolboxes*. Они очень важны для большинства пользователей MATLAB, так как позволяют изучать и применять специализированные методы. *Toolboxes* - это всесторонняя коллекция функций MATLAB (М-файлов), которые позволяют решать частные классы задач. *Toolboxes* применяются для обработки сигналов, систем контроля, нейронных сетей, нечеткой логики, вэйвлетов, моделирования и т.д.

Система MATLAB состоит из пяти основных частей.

**Язык MATLAB.** Это язык матриц и массивов высокого уровня с управлением потоками, функциями, структурами данных, вводом-выводом и особенностями объектно-ориентированного программирования.

**Среда MATLAB.** Это набор инструментов и приспособлений, с которыми работает пользователь или программист MATLAB. Она включает в себя средства для управления переменными в рабочем пространстве MATLAB, вводом и выводом данных, а также создания, контроля и отладки М-файлов и приложений MATLAB.

**Управляемая графика.** Это графическая система MATLAB, которая включает в себя команды высокого уровня для визуализации двух- и трехмерных данных, обработки изображений, анимации и иллюстрированной графики. Она также включает в себя команды низкого уровня, позволяющие полностью редактировать внешний вид графики, также как при создании Графического Пользовательского Интерфейса (GUI) для MATLAB приложений.

**Библиотека математических функций.** Это обширная коллекция вычислительных алгоритмов от элементарных функций, таких как сумма, синус, косинус, комплексная арифметика, до более сложных, таких как обращение матриц, нахождение собственных значений, функции Бесселя, быстрое преобразование Фурье.

**Программный интерфейс.** Это библиотека, которая позволяет писать программы на Си и Фортране, которые взаимодействуют с MATLAB. Она включает средства для вызова программ из MATLAB (динамическая связь), вызывая MATLAB как вычислительный инструмент и для чтения-записи MAT-файлов.

**Simulink,** сопутствующая MATLAB программа, - это интерактивная система для моделирования нелинейных динамических систем. Она представляет собой среду, управляемую мышью, которая позволяет моделировать процесс путем перетаскивания блоков диаграмм на экране и их манипуляцией. Simulink работает с линейными, нелинейными, непрерывными, дискретными, многомерными системами.

**Blocksets** - это дополнения к Simulink, которые обеспечивают библиотеки блоков для специализированных приложений, таких как связь, обработка сигналов, энергетические системы.

**Real-Time Workshop** - это программа, которая позволяет генерировать С код из блоков диаграмм и запускать их на выполнение на различных системах реального времени.

## **Матрицы**

Лучший способ начать работу с MATLAB — это научиться обращаться с матрицами. В этой главе мы покажем вам, как надо это делать. В MATLAB матрица - это прямоугольный массив чисел. Особое значение придается матрицам 1x1, которые являются скалярами, и матрицам, имеющим один столбец или одну строку, — векторам. MATLAB использует различные способы для хранения численных и не численных данных, однако вначале лучше всего рассматривать все данные как матрицы. MATLAB организован так, чтобы все операции в нем были как можно более естественными. В то время как другие программные языки работают с числами как элементами языка, MATLAB позволяет вам быстро и легко оперировать с целыми матрицами.

### Ввод матриц

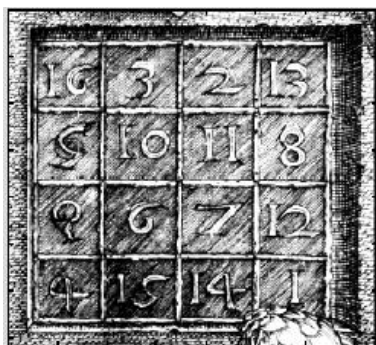
Вы можете вводить матрицы в MATLAB несколькими способами:

вводить полный список элементов

загружать матрицы из внешних файлов

генерировать матрицы, используя встроенные функции

создавать матрицы с помощью ваших собственных функций в М-файлах



Начнем с введения магической матрицы Дюрера (рис. 1) как списка элементов. Вы должны следовать нескольким основным условиям:

отделять элементы строки пробелами или запятыми

использовать точку с запятой ; для обозначения окончания каждой строки

окружать весь список элементов квадратными скобками, [ ].

Чтобы ввести матрицу Дюрера просто напишите:

```
A = [16 3 2 13; 5 10 11 8; 9 6 7 12; 4 15 14 1]
```

MATLAB отобразит матрицу, которую мы ввели,

```
A =
```

```
16 3 2 13
```

5 10 11 8

9 6 7 12

4 15 14 1

Если мы ввели матрицу, то она автоматически запоминается средой MATLAB. И мы можем к ней легко обратиться как к  $A$ . Сейчас, когда мы имеем  $A$  в рабочем пространстве MATLAB, посмотрим, что делает её такой интересной. Почему она называется магической?

Операции суммирования элементов, транспонирования и диагонализации матрицы

Вы возможно уже знаете, что особые свойства магического квадрата связаны с различными способами суммирования его элементов. Если вы берёте сумму элементов вдоль какой-либо строки или столбца, или вдоль какой-либо из двух главных диагоналей, вы всегда получите одно и то же число. Давайте проверим это, используя MATLAB. Первое утверждение, которое мы проверим -

$sum(A)$

MATLAB выдаст ответ

$ans =$

34 34 34 34

Когда выходная переменная не определена, MATLAB использует переменную  $ans$ , коротко от  $answer$  - ответ, для хранения результатов вычисления. Мы подсчитали вектор-строку, содержащую сумму элементов столбцов матрицы  $A$ . Действительно, каждый столбец имеет одинаковую сумму, магическую сумму, равную 34.

А как насчет сумм в строках? Лучший способ получить сумму в строках - это транспонировать нашу матрицу, подсчитать сумму в столбцах, а потом транспонировать результат. Операция транспонирования обозначается апострофом или одинарной кавычкой. Она зеркально отображает матрицу относительно главной диагонали и меняет строки на столбцы. Таким образом

$sum(A)'$

вызывает результат вектор-столбец, содержащий суммы в строках

$ans = 34$

34

34

34



Сумму элементов на главной диагонали можно легко получить с помощью функции `diag`, которая выбирает эту диагональ.

```
diag(A)
```

```
ans = 16
```

```
10
```

```
7
```

```
1
```

A функция

```
sum(diag(A)) вызывает
```

```
ans = 34
```

Другая диагональ, называемая антидиагональю, не так важна математически, поэтому MATLAB не имеет специальной функции для неё. Но функция, которая вначале предполагалась для использования в графике, `fliplr`, зеркально отображает матрицу слева направо.

```
sum(diag(fliplr(A)))
```

```
ans = 34
```

Индексы

Элемент в строке  $i$  и столбце  $j$  матрицы  $A$  обозначается  $A(i,j)$ . Например,  $A(4,2)$  - это число в четвертой строке и втором столбце. Для нашего магического квадрата  $A(4,2) = 15$ . Таким образом, можно вычислить сумму элементов в четвертом столбце матрицы  $A$ , набрав

```
A(1,4) + A(2,4) + A(3,4) + A(4,4)
```

получим

```
ans = 34
```

Однако это не самый лучший способ суммирования отдельной строки.

Также возможно обращаться к элементам матрицы через один индекс,  $A(k)$ . Это обычный способ ссылаться на строки и столбцы матрицы. Но его можно использовать только с двумерными матрицами. В этом случае массив рассматривается как длинный вектор, сформированный из столбцов исходной матрицы.

Так, для нашего магического квадрата,  $A(8)$  - это другой способ ссылаться на значение 15, хранящееся в  $A(4,2)$ .

Если вы пытаетесь использовать значение элемента вне матрицы, MATLAB выдаст ошибку:

```
t=A(4,5)
```

??? *Index exceeds matrix dimensions.*

С другой стороны, если вы сохраняете значение вне матрицы, то размер матрицы увеличивается.

$X=A;$

$X(4,5) = 17$

$X =$

$16\ 3\ 2\ 13\ 0$

$5\ 10\ 11\ 8\ 0$

$9\ 6\ 7\ 12\ 0$

$4\ 15\ 14\ 1\ 17$

Оператор двоеточия

Двоеточие : - это один из наиболее важных операторов MATLAB. Он проявляется в различных формах. Выражение

$1:10$

- это вектор-строка, содержащая целые числа от 1 до 10

$123456789\ 10$

Для получения обратного интервала, опишем приращение. Например

$100:-7:50$

что дает

$100\ 93\ 86\ 79\ 72\ 65\ 58\ 51$

Или

$0:\pi/4:\pi$

что даст

$0\ 0.7854\ 1.5708\ 2.3562\ 3.1416$

Индексное выражение, включая двоеточие, относится к части матрицы.

$A(1:k, j)$

это первые k элементов j -го столбца матрицы A.

Так

$sum(A(1:4,4))$

вычисляет сумму четвертой строки. Но есть и лучший способ. Двоеточие, само по себе, обращается ко всем элементам в строке и столбце матрицы, а слово end — к последней строке или столбцу. Так

```
sum (A (:, end) )
```

вычисляет сумму элементов в последнем столбце матрицы A

```
ans = 34
```

### Выражения

Как и большинство других языков программирования, MATLAB предоставляет возможность использования математических выражений, но в отличие от многих из них, эти выражения в MATLAB включают матрицы. Основные составляющие выражения:

переменные

числа

операторы

функции

Переменные

В MATLAB нет необходимости в определении типа переменных или размерности. Когда MATLAB встречает новое имя переменной, он автоматически создает переменную и выделяет соответствующий объем памяти. Если переменная уже существует, MATLAB изменяет ее состав и если это необходимо выделяет дополнительную память. Например,

```
num_students = 25
```

создает матрицу 1x1 с именем num\_students и сохраняет значение 25 в ее единственном элементе.

Имена переменных состоят из букв, цифр или символов подчеркивания. MATLAB использует только первые 31 символ имени переменной. MATLAB чувствителен к регистрам, он различает заглавные и строчные буквы. Поэтому A и a - не одна и та же переменная. Чтобы увидеть матрицу связанную с переменной, просто введите название переменной.

### Числа

MATLAB использует принятую десятичную систему счисления, с необязательной десятичной точкой и знаками плюс-минус для чисел. Научная система счисления использует букву e для определения множителя степени десяти. Мнимые числа используют i или j как суффикс. Некоторые примеры правильных чисел приведены ниже

```
3 -99 0.0001
```

```
9.6397238 1.60210e-20 6.02252e23
```

```
1i -3.14159j 3e5i
```

Числа с плавающей точкой обладают ограниченной точностью - приблизительно 16 значащих цифр и ограниченным диапазоном - приблизительно от 10-308 до 10308

### Операторы

Выражения используют обычные арифметические операции и правила старшинства.

+ сложение

- вычитание

\* умножение

/ деление

Л степень

' комплексно сопряженное транспонирование

() определение порядка вычисления

#### 2.3 Специальные символы

[ ] – квадратные скобки используют для создания матриц и векторов;

– пробел служит для разделения элементов матриц;

, – запятая применяется для разделения элементов матриц и операторов в строке ввода;

; – точка с запятой отделяет строки матриц, а точка с запятой в конце оператора (команды) отменяет вывод результата на экран;

: – двоеточие используется для указания диапазона (интервала изменения величины) и в качестве знака групповой операции над элементами матриц;

% – знак процента обозначает начало комментария;

! – отмечает начало команды DOS

' – апостроф указывает на символьные строки.

#### 2.4 Функции

MATLAB предоставляет большое количество элементарных математических функций, таких как `abs`, `sqrt`, `exp`, `sin`. Вычисление квадратного корня или логарифма отрицательного числа не является ошибкой: в этом случае результатом является соответствующее комплексное число. MATLAB также предоставляет и более сложные функции, включая Гамма функцию и функции Бесселя. Большинство из этих функций имеют комплексные аргументы. Чтобы вывести список всех элементарных математических функций, наберите

*help elfun*

Для вывода более сложных математических и матричных функций, наберите

*help specfun*

*help elmat*

соответственно.

Некоторые функции, такие как `sqrt` и `sin`, - встроенные. Они являются частью MATLAB, поэтому они очень эффективны, но их вычислительные детали трудно доступны. В то время как другие функции, такие как `gamma` и `sinp`, реализованы в М-файлах. Поэтому вы можете легко увидеть их код и, в случае необходимости, даже модифицировать его.

Несколько специальных функций предоставляют значения часто используемых констант.

`pi` 3.14159265...

`i` мнимая единица,  $\sqrt{-1}$

`j` то же самое, что и `i`

`realmin` наименьшее число с плавающей точкой, 2-1022

`realmax` наибольшее число с плавающей точкой, (2-е)21023

`Inf` бесконечность

`NaN` не число

Бесконечность появляется при делении на нуль или при выполнении математического выражения, приводящего к переполнению, т.е. к превышению `realmax`. Не число (`NaN`) генерируется при вычислении выражений типа `0/0` или `Inf-Inf`, которые не имеют определенного математического значения.

Имена функций не являются зарезервированными, поэтому возможно изменять их значения на новые, например

*eps = 1.e-6*

и далее использовать это значение в последующих вычислениях. Начальное значение может быть восстановлено следующим образом

*clear eps*

### 3. Графика

MATLAB имеет широкие возможности для графического изображения векторов и матриц, а также для создания комментариев и печати графики. Эта глава описывает несколько наиболее важных графических функций и дает примеры их применения.

### 3.1 Создание графика

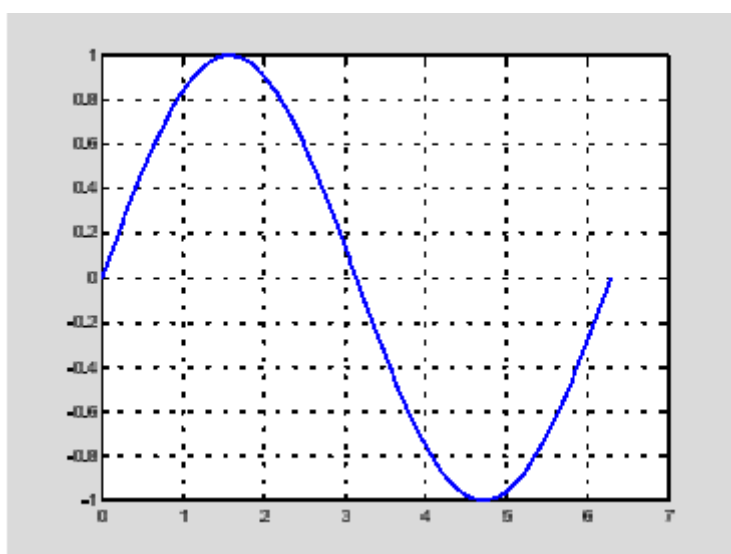
Функция `plot` имеет различные формы, связанные с входными параметрами, например `plot(y)` создает кусочно-линейный график зависимости элементов `y` от их индексов. Если вы задаете два вектора в качестве аргументов, `plot(x,y)` создаст график зависимости `y` от `x`.

Например, для построения графика значений функции `sin` от нуля до  $2\pi$ , сделаем следующее

```
t = 0:pi/100:2*pi;
```

```
y = sin(t);
```

```
plot(t,y)
```

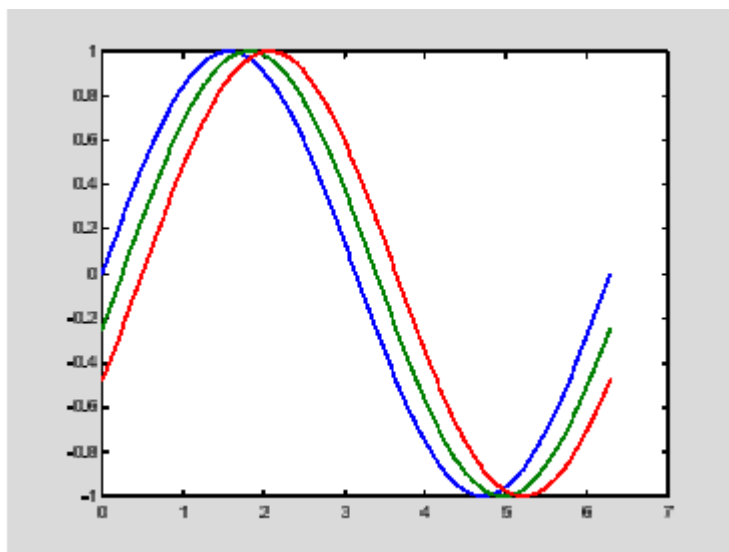


Вызов функции `plot` с многочисленными парами `x-y` создает многочисленные графики. MATLAB автоматически присваивает каждому графику свой цвет (исключая случаи, когда это делает пользователь), что позволяет различать заданные наборы данных. Например, следующие три строки отображают график близких функций, и каждой кривой соответствует свой цвет:

```
y2 = sin(t-.25);
```

```
y3 = sin(t-.5);
```

```
plot( t, y, t, y2, t, y3)
```



Возможно изменение цвета, стиля линий и маркеров, таких как знаки плюс или кружки, следующим образом

```
plot(x, y, 'цвет_стиль_маркер')
```

цвет\_стиль\_маркер это 1-, 2-, 3-х символьная строка (заключенная в одинарные кавычки), составленная из типов цвета, стиля линий и маркеров:

Символы, относящие к цвету: 'c', 't', 'y', 'r', 'g', 'b', 'w' и 'k'. Они обозначают голубой, малиновый, желтый, красный, зеленый, синий, белый и черный цвета соответственно.

Символы, относящиеся к типу линий: '-' для сплошной, '-' для разрывной, ':' для пунктирной, '-.' для штрихпунктирной линий и 'попе' для её отсутствия.

Наиболее часто встречающиеся маркеры '+','o','\*','x'.

Например, выражение

```
plot(x,y,'y:+')
```

строит желтый пунктирный график и помещает маркеры '+' в каждую точку данных. Если вы определяете только тип маркера, но не определяете тип стиля линий, то MATLAB выведет только маркеры.

### Окна изображений

Функция plot автоматически открывает новое окно изображения (далее окно), если до этого его не было на экране. Если же оно существует, то plot использует его по умолчанию. Для открытия нового окна и выбора его по умолчанию, наберите

*figure*

Для того, чтобы сделать существующее окно текущим -

*figure(n)*

где n - это номер в заголовке окна. В этом случае результаты всех последующих команд будут выводиться в это окно.

Добавление кривых на существующий график

Команда `hold` позволяет добавлять кривые на существующий график. Когда вы набираете

*hold on*

MATLAB не стирает существующий график, а добавляет в него новые данные, изменяя оси, если это необходимо. Например, следующий элемент кода вначале создает контурные линии функции `peaks`, а затем накладывает псевдоцветной график той же функции:

*[x,y,z] = peaks;*

*contour(x,y,z,20,'k')*

*hold on*

*pcolor(x,y,z) shading interp*

Команда `hold on` является причиной того, что график `pcolor` комбинируется с графиком `contour` в одном окне

Подграфики

Функция `subplot` позволяет выводить множество графиков в одном окне или распечатывать их на одном листе бумаги.

*subplot(m,n,p)*

разбивает окно изображений на матрицу m на n подграфиков и выбирает n-ый подграфик текущим. Графики нумеруются вдоль первого в верхней строке, потом во второй и т.д. Например, для того, чтобы представить графические данные в четырех разных подобластях окна необходимо выполнить следующее:

*t = 0:pi/10:2\*pi;*

*[C,X,Z] = cylinder(4\*cos(t));*

*subplot(2,2,1)*

*mesh(X)*

*subplot(2,2,2); mesh(Y)*

*subplot(2,2,3); mesh(Z)*



```
subplot(2,2,4); mesh(X,Y,Z)
```

### 3.3 Управление осями

Функция `axis` имеет несколько возможностей для настройки масштаба, ориентации и коэффициента сжатия.

Обычно MATLAB находит максимальное и минимальное значение и выбирает соответствующий масштаб и маркирование осей. Функция `axis` заменяет значения по умолчанию предельными значениями, вводимыми пользователем.

```
axis( [xmin xmax ymin ymax] )
```

В функции `axis` можно также использовать ключевые слова для управления внешним видом осей. Например

```
axis square
```

*создает ч и у оси равной длины, а*

```
axis equal
```

*создает отдельные отметки приращений для ч и у осей одинаковой длины. Так функция*

```
plot(exp(i*t))
```

*следующая либо за axis square, либо за axis equal превращает овал в правильный круг.*

```
axis auto
```

*возвращает значения по умолчанию и переходит в автоматический режим.*

```
axis on
```

*включает обозначения осей и метки промежуточных делений.*

```
axis off
```

*выключает обозначения осей и метки промежуточных делений.*

```
grid off
```

*выключает сетку координат, а*

```
grid on
```

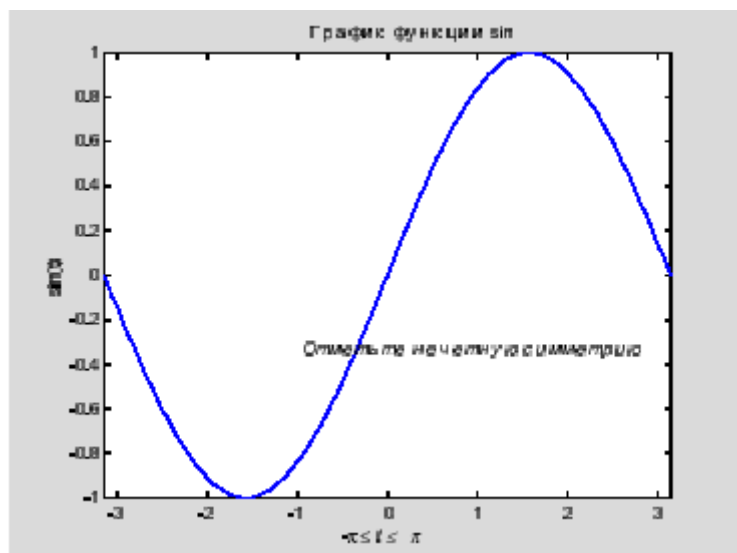
*включает её заново.*

### **Подписи к осям и заголовки**

Функции `xlabel`, `ylabel`, `zlabel` добавляют подписи к соответствующим осям, функция `title` добавляет заголовок в верхнюю часть окна, а функция `text` вставляет текст в любое место графика. Использование TEX представления

позволяет применять греческие буквы, математические символы и различные шрифты. Следующий пример демонстрирует эту возможность.

```
t = -pi:pi/100:pi;
y = sin(t);
plot(t,y)
axis([-pi pi -1 1])
xlabel( '-\pi \leq t \leq \pi ')
ylabel( 'sin(t) ')
title( 'График функции sin ')
text(-1, -1/3, '\it{отметьте нечетную симметрию}')
```



### Функции mesh и surface

MATLAB определяет поверхность как ж координаты точек над координатной сеткой плоскости x-y, используя прямые линии для соединения соседних точек. Функции mesh и surface отображают поверхность в трех измерениях. При этом

mesh создает каркасную поверхность, где цветные линии соединяют только заданные точки, а функция surface вместе с линиями отображает в цвете и саму поверхность.

### Визуализация функций двух переменных

Для отображения функции двух переменных,  $z = f(x,y)$ , создаются матрицы X и Y, состоящие из повторяющихся строк и столбцов соответственно, перед использованием функции. Затем используют эти матрицы для вычисления и отображения функции. Функция meshgrid преобразует область определения, заданную через один вектор или два вектора x и y, в матрицы X и Y для использования при вычислении функции

двух переменных. Строки матрицы  $X$  дублируют вектор  $x$ , а столбцы  $X$  - вектор  $y$ .

Для вычисления двумерной функции  $\text{sinc}$ ,  $\sin(r)/r$ , в области  $x$ - $y$  поступают следующим образом

```
[Y, X] = meshgrid(-8:.5:8);
R = sqrt(X.^2+Y.^2)+eps;
Ж = sin(R)./R; mesh (Y, X, Ж)
```

В этом примере  $R$  - это расстояние от начала координат, которому соответствует центр матрицы. Добавление  $\text{eps}$  позволяет избежать неопределенности  $0/0$  в начале координат.

### Изображения

Двумерные массивы могут отображать как изображения, где элементы массива определяют их яркость и цвет. Например

```
load durer whos
```

покажет, что файл `durer.mat` в директории `demo` состоит из матрицы размером 648 на 509 (матрицы  $X$ ) и матрицы размером 128 на 3 (матрицы  $\text{map}$ ). Элементы матрицы  $X$  - это целые числа от 1 до 128, которые служат индикаторами в цветном отображении,  $\text{map}$ . Следующие строки

```
image(X)
colormap(map)
axis image
```

Воспроизводят гравюру Дюрера. Высокое разрешение магического квадрата, находящегося в правом верхнем углу, доступно в другом файле. Наберите

```
load detail
```

и используйте стрелку 'вверх' на клавиатуре для повторного запуска команд `image`, `colormap` и `axis`.

```
colormap(hot)
```

добавит цветовую гамму двадцатого века на гравюру шестнадцатого.

### Печать графики

Опция `Print` в меню `File` и команда `print` печатают графику `MATLAB`. Меню `Print` вызывает диалоговое окно, которое позволяет выбирать общие стандартные варианты печати. Команда `print` обеспечивает большую гибкость при выводе выходных данных и позволяет контролировать печать из `M`-файлов. Результат может быть послан прямо на принтер, выбранный по

умолчанию, или сохранен в заданном файле. Возможно широкое варьирование формата выходных данных, включая использование PostScript.

Например, следующая команда сохранит текущее окно изображения как цветной PostScript Level 2 Encapsulated в файле `magic-square.eps`:

```
print -depsc2 magic-square.eps
```

Важно знать возможности вашего принтера перед использованием команды `print`. Например, файлы Postscript Level 2 обычно меньше и воспроизводятся намного быстрее, чем Postscript Level 1. Однако, не все Postscript принтеры поддерживают Level 2, таким образом вам необходимо узнать, что может обрабатывать ваше устройство вывода. MATLAB использует дифференцированный подход для вывода графики и текста, даже для черно-белых устройств.

### Командное окно

До сих пор, мы использовали только командную строку MATLAB, печатая команды и выражения и наблюдая результаты. В этой главе описано несколько способов изменения внешнего вида командного окна. Если ваша система позволяет вам выбирать шрифт, то мы рекомендуем использовать шрифты с фиксированной шириной, такие как `Fixedsys` или `Courier`, для обеспечения правильного межстрочного интервала.

Если самый большой элемент матрицы больше 103 или самый маленький меньше 10<sup>-3</sup>, MATLAB применяет общий масштабный коэффициент для форматов `short` и `long`. В дополнение к командам `format`, рассмотренным выше

```
format compact
```

убирает много пустых линий, появляющихся на выходе. Это позволяет вам видеть больше информации на экране. Если вы хотите изменить контроль над форматом выходных данных, используйте функции `sprint/` и `/print/`.

Сокращение выходных данных

Если вы наберете выражение и нажмете Return или Enter, MATLAB автоматически выведет результат на экран. Однако если в конце строки вы поставите точку с запятой, MATLAB проведет вычисления, но не отобразит их. Это часто бывает нужно при создании больших матриц. Например,

```
A = magic(100);
```

Длинные командные строки

Если выражение не помещается на одной строке, используйте троеточие, а за ним Return или Enter, для обозначения того, что выражение продолжается на следующей строке. Например

$$s = 1 - 1/2 + 1/3 - 1/4 + 1/5 - 1/6 + 1/7 \dots - 1/8 + 1/9 - 1/10 + 1/11 - 1/12;$$

Пробелы вокруг знаков =, +, - не обязательны, но улучшают читаемость текста.

### Редактор командной строки

Различные стрелки и управляющие клавиши на вашей клавиатуре позволяют вам вызывать, редактировать и многократно использовать команды, набранные ранее. Например, предположим, что вы допустили ошибку при вводе

$$rho = (1 + sqrt(5))/2$$

Вы ошиблись в написании sqrt. MATLAB ответит вам предупреждением *Undefined function or variable 'sqrt'*.

Вместо того, чтобы заново набирать всю строку, просто нажмите клавишу T. Тогда на экране изобразится ошибочная команда. Используйте клавишу ← для перемещения курсора и вставки пропущенной буквы r. Повторное использование клавиши T вызовет предыдущие строки. Наберите несколько символов, и тогда клавиша T найдет предыдущую строку, которая начинается с них.

## Среда MATLAB

Среда MATLAB включает в себя как совокупность переменных, созданных за время сеанса работы MATLAB, так и набор файлов, содержащих программы и данные, которые продолжают существовать между сеансами работы.

### Рабочее пространство

Рабочее пространство - это область памяти, доступная из командной строки MATLAB. Две команды, who и whos, показывают текущее содержание рабочего пространства. Команда who выдает краткий список, а команда whos размер и используемую память.

Ниже представлен вывод, осуществленный командой whos, на рабочем пространстве, содержащем результаты из некоторых примеров этой книги. Он показывает различия в структуре данных MATLAB. В качестве упражнения попробуйте найти сегмент кода, который соответствует каждой из ниже приведенных переменных.

*whos*

*Name Size Bytes Class*

*A 4x4 128 double array*

*D 5x3 120 double array*

*M 10x1 3816 cell array*

*S 1x3 442 struct array*

*h 1x11 22 char array*

*z 1x1 8 double array*

*s 1x5 10 char array*

*n 2x5 20 char array*

*Grand total is 471 elements using 4566 bytes.*

Для удаления всех существующих переменных из рабочего пространства MATLAB, введите

*clear*

### **Команда save**

Команда `save` сохраняет содержание рабочего пространства в MAT-файле, который может быть прочитан командой `load` в последующих сеансах работы MATLAB. Например,

*save August17th*

сохраняет содержание всего рабочего пространства в файле `August17th.mat`. Если нужно, вы можете сохранить только определенные переменные, указывая их имена после имени файла.

### **Справка и текущая документация**

Есть несколько способов получить текущую документацию по функциям MATLAB.

Команда `help`

Окно справки

MATLAB Help Desk

Текущие справочные страницы

Связь с The MathWorks, Inc.

Команда `help`

Команда `help` - это самый основной способ определения синтаксиса и поведения отдельных функций. Информация отображается прямо в командном окне. Например

*help magic*

выдаст

*MAGIC Magic square.*

MAGIC(H) is an N-by-N matrix constructed from the integers 1 through H\*2 with equal row, column, and diagonal sums. Produces valid magic squares for N = 1,3,4,5,...

Замечание MATLAB в текущей справке использует заглавные буквы для функций и переменных для того, чтобы выделить их из текста. Однако, при наборе имен функций всегда используйте соответствующие строчные буквы, так как MATLAB чувствителен к регистрам, а все имена функции строчные.

Все функции MATLAB организованы в логические группы и структура директорий MATLAB базируется на этом группировании. Например, все функции линейной алгебры находятся в директории `matfun`. Чтобы вывести имена всех функций в этой директории с кратким описанием, надо набрать

```
help matfun
```

```
Matrix functions - numerical linear algebra.
```

```
Matrix analysis.
```

```
norm - Matrix or vector norm,
```

```
normest - Estimate the matrix 2-norm.
```

Команда `lookfor`

Команда `lookfor` позволяет искать функции по ключевому слову. Она просматривает первую строку текста справки, называемую строкой H1, для каждой функции MATLAB и возвращает строки H1, содержащие заданное ключевое слово. Например, MATLAB не имеет функции с именем `inverse`. Поэтому ответ на запрос

```
help inverse
```

```
будет
```

```
inverse.m not found.
```

В то время как

```
lookfor inverse
```

найдет множество согласованных ответов. В зависимости от того, какие `toolboxes` вы установили, вы получите соответствующие записи. Например

```
INVHILB Inverse Hubert matrix.
```

```
ACOS Inverse cosine.
```

```
ACOSH Inverse hyperbolic cosine.
```

```
ACOT Inverse cotangent.
```

```
ACOTH Inverse hyperbolic cotangent.
```

```
ACSC Inverse cosecant.
```

*ACSCH Inverse hyperbolic cosecant.*

*ASEC Inverse secant.*

*ASECH Inverse hyperbolic secant.*

*ASIN Inverse sine.*

*ASINH Inverse hyperbolic sine.*

*ATAN Inverse tangent.*

Добавление ключа *-all* в команду *lookfor*, как, например,

*lookfor -all*

позволяет искать ключевое слово во всех записях справки, а не только в строке H1.

Печать текущих справочных страниц

Версии текущих справочных страниц, как и большинство документации, также доступно в формате PDF (Portable Document Format) через Help Desk. Эти страницы обрабатываются с помощью Adobe's Acrobat reader. Они воспроизводят внешний вид страниц после печати, полностью с шрифтами, графикой, с заданным форматом и рисунками. Это лучший способ получить печатные копии справочных материалов.

Материал подготовлен на основе:

Getting Started with MATLAB, pdf документ, перевод с английского  
Конюшенко В.В.



## **Multisim**

### **Краткие сведения**

#### **1. Введение**

Электронная система моделирования Multisim имитирует реальное рабочее место исследователя – лабораторию, оборудованную измерительными приборами, работающими в реальном масштабе времени. С ее помощью можно создавать, моделировать как простые, так и сложные аналоговые и цифровые радиофизические устройства.

В настоящей лабораторной работе описываются основные принципы работы с электронной системой моделирования Multisim 9. Для четкого понимания принципов ее работы необходимо:

- знание основных принципов работы операционной системы Windows;
- понимание принципов работы основных измерительных приборов (осциллограф, мультиметр, и т.п.);
- знание отдельных элементов радиоэлектронных устройств.

#### **2. Основные принципы создания схемы.**

Работа с электронной системой моделирования Multisim включает в себя три основных этапа: создание схемы, выбор и подключение измерительных приборов, и, наконец, активация схемы – расчет процессов, протекающих в исследуемом устройстве.

В общем случае процесс создания схемы начинается с размещения на рабочем поле Multisim компонентов из библиотеки программы. Подразделы библиотеки программы Multisim поочередно могут быть вызваны с помощью иконок, расположенных на панели инструментов (рис. 1). Каталог выбранного раздела библиотеки располагается в вертикальном окне справа или слева от рабочего поля (устанавливается в любое место перетаскиванием стандартным способом – за шапку заголовка). Для выбора требуемого элемента из библиотеки необходимо подвести курсор мыши к соответствующей иконке и нажать один раз на стрелку раскрывающегося списка, после чего выбрать в списке необходимый для работы элемент. После этого необходимый для создания схемы значок (символ) компонента переносится на рабочее поле программы нажатием левой клавиши мыши. При размещении компонентов схемы на рабочем поле программы можно также воспользоваться контекстным меню, возникающим при нажатии на правую клавишу мыши на свободном месте рабочего поля. На этом этапе необходимо предусмотреть место для размещения контрольных точек и иконок контрольно-измерительных приборов.

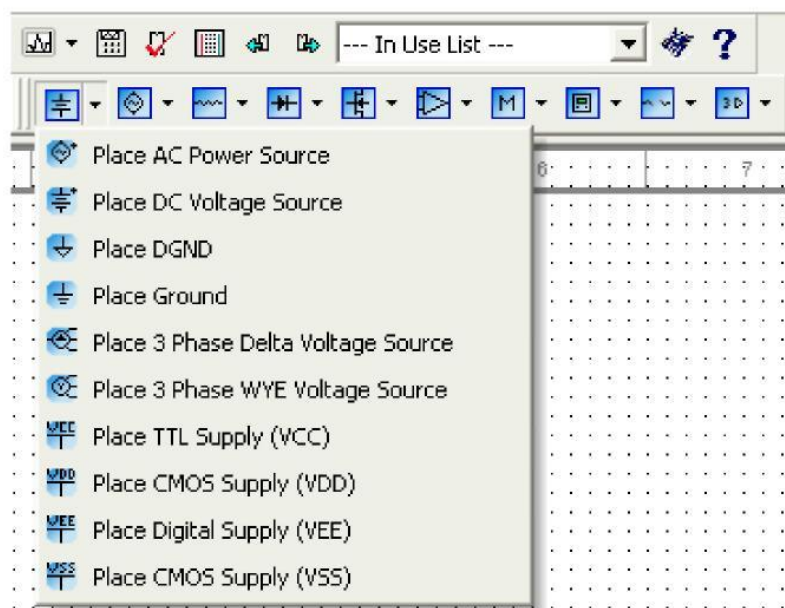


Рис. 1. Каталоги библиотеки компонентов Multisim 9

Выделенный компонент схемы (выделяется рамкой из штриховой синей линии) можно повернуть (контекстного меню, кнопок на панели инструментов или пункта меню *Circuit>Rotate*) или зеркально отразить относительно вертикальной (горизонтальной) оси (команда меню *Circuit>Flip Vertical (Horizontal)*, контекстное меню, кнопки на панели инструментов). При повороте большинство компонентов поворачиваются на  $90^\circ$  против часовой стрелки при каждом выполнении команды, для измерительных приборов (амперметр, вольтметр и др.) меняются местами клеммы подключения.

В готовой схеме пользоваться поворотом и отражением элементов нецелесообразно, поскольку это чаще всего приводит к путанице соединительных проводов - в этом случае компонент нужно отключить от цепи, и только потом вращать (отражать).

По умолчанию устанавливается виртуальный элемент, обладающий идеальными свойствами (например, отсутствие внутренних шумов и потерь) того или иного элемента. С помощью двойного щелчка по значку компонента можно изменить его свойства. В раскрывающемся диалоговом окне устанавливаются требуемые параметры (как правило, номинал элемента схемы и ряд других параметров для других элементов типа измерительных приборов или сложных интегральных схем) и выбор подтверждается нажатием кнопки «Ok» или клавиши «Enter» на клавиатуре. В том же диалоговом окне, при нажатии кнопки *Replace* появляется диалоговое окно с указанием всей библиотеки элементов. С помощью этого окна можно заменить идеальный элемент его реальным аналогом, при этом варьируется не только его номинал, но и производитель конкретных схемных элементов, а также серия элемента. Для большого числа компонентов можно выбрать

параметры, соответствующие реальным элементам (диодам, транзисторам и т.п.) различных производителей.

При создании схем удобно также пользоваться динамическим меню, которое вызывается нажатием правой кнопки мыши. Меню содержит команды Help (помощь), Paste (вставить), Zoom In (увеличить), Zoom Out (уменьшить), Schematic Options (параметры схемы), а также команды Add <Название компонента>. Эта команда позволяет добавить на рабочее поле компоненты, не обращаясь к каталогам библиотеки. Количество команд Add <Название компонента> в списке меню определяется количеством типов компонент (резисторов, знака заземления и т.д.), уже имеющихся на рабочем поле.

После размещения компонентов производится соединение их выводов проводниками. При этом необходимо учитывать, что к выводу компонента можно подключить только один проводник. Для выполнения подключения курсор мыши подводится к выводу компонента, и после появления площадки, нажимается левая кнопка мыши. Появляющийся при этом проводник протягивается к выводу другого компонента до появления на нем такой же площадки, после чего ещё раз нажимается левая кнопка мыши. При необходимости подключения к этим выводам других проводников в контекстном меню (появляется при нажатии правой кнопки мыши) выбирается точка (символ соединения, обозначен как Junction) и переносится на ранее установленный проводник. Если на ней виден след от пересекающего проводника, то электрического соединения нет и точку необходимо установить заново. После удачной установки к точке соединения можно подключить еще два проводника. Если соединение нужно разорвать, курсор подводится к соответствующему проводу и выделяется левой кнопкой мыши, после чего нажимается клавиша Delete.

Если необходимо подключить вывод к имеющемуся на схеме проводнику, то проводник от вывода компонента курсором подводится к указанному проводнику и после появления точки соединения нажимается левая кнопка мыши. Следует отметить, что прокладка соединительных проводников производится автоматически, причем препятствия - компоненты и другие проводники - огибаются по ортогональным направлениям (по горизонтали или вертикали).

Подключение к схеме контрольно-измерительных приборов производится аналогично. Панель с контрольно-измерительным оборудованием (за исключением амперметра и вольтметра) расположена вертикально с правой стороны рабочей области, и включает в себя такие элементы как мультиметр, осциллограф (2-х и 4-х канальный), ваттметр, функциональный генератор, бодиplotтер, спектральный анализатор и т.д. Более подробно работа некоторых из этих приборов будет описана ниже.




Для таких приборов, как осциллограф или логический анализатор, соединения целесообразно проводить цветными проводниками, поскольку их цвет определяет цвет соответствующей осциллограммы.

Каждый элемент может быть передвинут на новое место. Для этого он должен быть выделен и перетащен с помощью мышки. При этом расположение соединительных проводов изменится автоматически. Можно также переместить целую группу элементов: для этого их нужно последовательно выделять мышкой при нажатой клавише Ctrl, а затем перетащить их в новое место. Если необходимо переместить отдельный сегмент проводника, к нему подводится курсор, нажимается левая кнопка и, после появления в вертикальной или горизонтальной плоскости двойного курсора, производятся нужные перемещения.

После подготовки схемы рекомендуется составить ее описание (соответствующее окно вызывается из меню Tools > Description box editor).

### **3. Описание основных элементов**

Как уже говорилось, в электронной системе Multisim имеется несколько разделов библиотеки компонентов, которые могут быть использованы при моделировании. Ниже приводится краткая справка по основным (естественно, не всем) компонентам. После названия в скобках приведены некоторые параметры компонента, которые могут быть изменены пользователем. Все компоненты условно разделим на ряд подгрупп.

	Батарея (напряжение). Длинная полоска соответствует положительной клемме.		
	Заземление (метка).		Источник постоянного тока (ток).
	Источник переменного синусоидального напряжения (эффективное значение напряжения, частота, фаза).		Источник переменного синусоидального тока (эффективное значение тока, частота, фаза).
	Источники фиксированного напряжения. Применяются в логических схемах.		Генератор однополярных прямоугольных импульсов (амплитуда, частота, коэффициент заполнения).
	Генератор амплитудно-модулированных колебаний (напряжение и частота несущей, коэффициент и частота модуляции).		Генератор фазо-модулированных колебаний (напряжение и частота несущей, индекс и частота модуляции).

**3.2. Пассивные элементы** (вкладка Basic) – библиотека, в которой собраны все пассивные компоненты, а также коммуникационные устройства.



Резистор (сопротивление).



Конденсатор (ёмкость).



Катушка индуктивности  
(индуктивность).



Трансформатор.



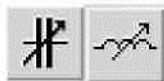
Реле (находится только в библиотеке элементов).



Переключатель, управляемый нажатием заданной клавиши (по умолчанию – пробел).



Потенциометр (реостат). Параметр «Key» определяет символ клавиши клавиатуры (по умолчанию A), при нажатии на которую сопротивление уменьшается на заданную в процентах величину (параметр «Increment», по умолчанию 5%) или увеличивается на такую же величину при нажатии клавиш Shift+«Key». Параметр «Setting» задает начальную установку сопротивления в процентах (по умолчанию – 50%), параметр «Resistance» задает номинальное значение сопротивления.



Конденсатор и катушка индуктивности переменной ёмкости. Действуют аналогично потенциометру.

**3.3. Полупроводниковые элементы** (Diode Components и Transistor Components) – диоды и транзисторы.



Полупроводниковый диод  
(тип).



Диод Шокли (тип)/



Стабилитрон (тип).



Тиристор или динистор (тип).



Светодиод (тип).




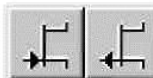



Симметричный динистор или  
диак (тип).



Выпрямительный мост (тип).



Симметричный тринистор или  
триак (тип).



	Биполярные $n-p-n$ и $p-n-p$ транзисторы, соответственно (тип).		Полевые транзисторы с управляющим $p-n$ переходом (тип).
	Полевые МОП-транзисторы с изолированным затвором ( $n$ -канальные с обогащенной подложкой и $p$ -канальные с обедненной подложкой), с отдельными или соединенными выводами подложки и истока (тип).		
	Полевые МОП-транзисторы с изолированным затвором ( $n$ -канальные с обогащенным затвором и $p$ -канальные с обедненным затвором), с отдельными или соединенными выводами подложки и истока (тип).		
	Арсенид-галлиевые $n$ - и $p$ -канальный полевые транзисторы (тип)		

#### 3.4. Аналоговые микросхемы (Analog Components).

	Линейная модель операционного усилителя (тип)		Разновидности операционных усилителей (тип)
	Нелинейная модель операционного усилителя (тип)		Компаратор напряжения

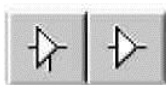
Указанные выше разделы библиотеки содержат основные схемотехнические элементы.

#### 3.5. Логические цифровые микросхемы (разделы библиотеки TTL и CMOS).

	Логические элементы И, И-НЕ (количество входов)		Логические элементы ИЛИ, ИЛИ-НЕ (количество входов)
	Логические элементы ИСКЛЮЧАЮЩЕЕ ИЛИ и		Логический элемент НЕ

### ИСКЛЮЧАЮЩЕЕ ИЛИ-НЕ

(количество входов)



Тростабильный буфер

(элемент с тремя

состояниями) и буфер (тип)



Триггер Шмидта (тип)

Более сложные элементы цифровой схемотехники (триггеры, мультиплексоры, декодеры и т.д.) не имеют в Multisim специальных обозначений и изображаются в виде пиктограммы (квадрат с различным числом выходов и соответствующими обозначениями). Определить тип того или иного схемного элемента можно по описанию в окне библиотеки.

### 3.6. Индикаторные устройства (Misc, Measurement Components или раздел Indicators в библиотеке).



Вольтметр с цифровым отсчетом (внутреннее сопротивление, режим измерения постоянного или переменного тока). Отрицательная клемма показана утолщенной черной линией.



Амперметр с цифровым отсчетом (внутреннее сопротивление, режим измерения постоянного или переменного тока). Отрицательная клемма показана утолщенной черной линией.



Лампа накаливания  
(напряжение, мощность).



Светоиндикатор (цвет свечения).



Семисегментный индикатор  
(тип).



Семисегментный индикатор с дешифратором (тип).



Линейка из десяти независимых светодиодов  
(напряжение, номинальный и минимальный ток).



Линейка из десяти светодиодов со встроенным АЦП (минимальное и минимальное напряжение).



Звуковой индикатор (частота звукового сигнала, напряжение и ток срабатывания).



### 3.14. Контрольно-измерительные приборы (панель Instruments).

Опишем некоторые представленные в Multisim контрольно-измерительные приборы.



Мультиметр.

Функциональный генератор.

Осциллограф.

Измеритель АЧХ и ФЧХ.

Панель контрольно-измерительных приборов содержит цифровой мультиметр, функциональный генератор, двухканальный осциллограф, измеритель амплитудно-частотных и частотных характеристик, генератор слов (кодированный генератор), 8-канальный логический анализатор и логический преобразователь, а также некоторые другие приборы (например, виртуальные мультиметр, функциональный генератор, осциллограф фирмы Agilent). Общий порядок работы с приборами такой: иконка прибора курсором мыши переносится на рабочее поле и подключается проводниками к исследуемой схеме. Для приведения прибора в рабочее (развернутое) состояние необходимо дважды щелкнуть курсором по его иконке.

**Мультиметр.** На лицевой панели мультиметра расположен дисплей для отображения результатов измерения, клеммы для подключения к схеме, и кнопки управления:

- выбор режима измерения тока, напряжения, сопротивления и ослабления (затухания);
- выбор режима измерения переменного или постоянного тока;
- режим установки параметров мультиметра.

После нажатия на кнопку установки параметров мультиметра открывается диалоговое окно, в котором можно выставить следующие параметры:

- Ammeter resistance — внутреннее сопротивление амперметра;
- Voltmeter resistance — входное сопротивление вольтметра;
- Ohmmeter current — ток через контролируемый объект;
- dB Relative value — установка эталонного напряжения  $V_1$  при



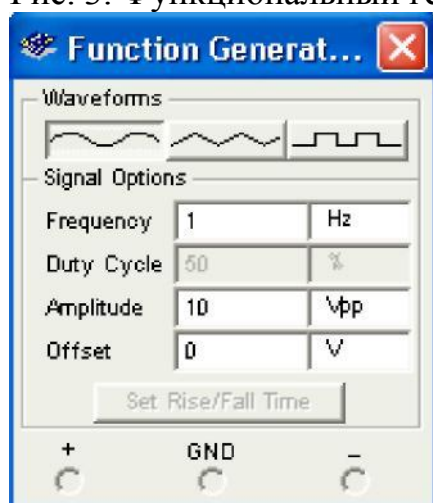
Рис. 2. Мультиметр

измерении ослабления или усиления в децибелах (по умолчанию  $V_1=774.597mV$ ). При этом для коэффициента передачи используется формула:  $K=20 \log(V_2/V_1)$ , где  $V_2$  – напряжение в контролируемой точке,  $K$  измеряется в децибеллах .

**Функциональный генератор.** Лицевая панель функционального генератора показана на рис. 3. Управление генератором осуществляется следующими органами управления:

- выбор формы выходного сигнала: синусоидальной (установлен по умолчанию), треугольной и прямоугольной;
- установка частоты выходного сигнала;
- установка коэффициента заполнения в %: для импульсных сигналов это отношение длительности импульса к периоду повторения – величина, обратная скважности, для треугольных сигналов – соотношение между длительностями переднего и заднего фронта;
- установка амплитуды выходного сигнала;
- установка смещения (постоянной составляющей) выходного сигнала;
- выходные зажимы; при заземлении клеммы COM (общий) клеммах "-" и "+" получается парафазный сигнал.

Рис. 3. Функциональный генератор



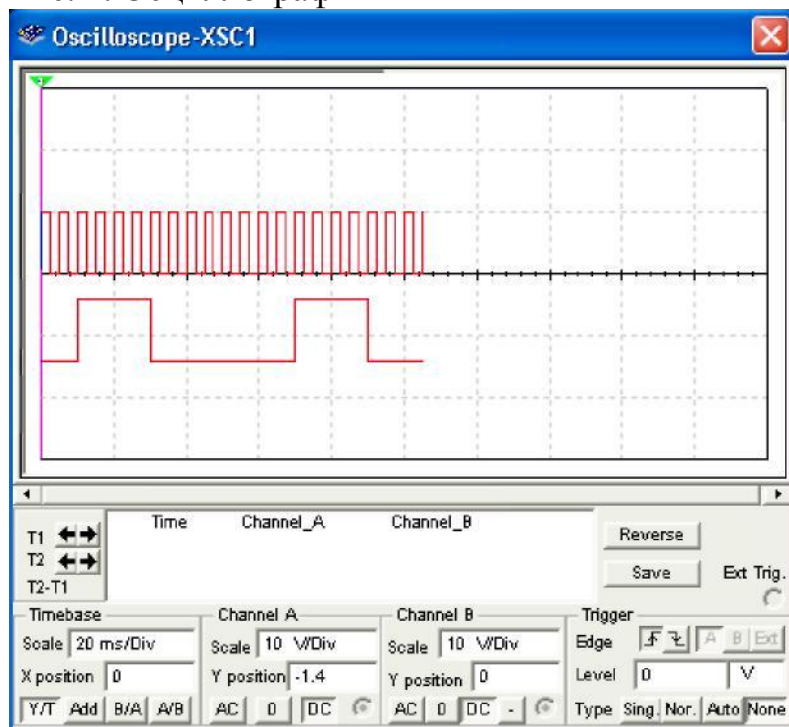
**Осциллограф.** Лицевая панель осциллографа показана на рис. 4. Осциллограф имеет два канала (CHANNEL) А и В (в случае двухканального осциллографа; в арсенале приборов Multisim имеется также и 4-х канальный осциллограф, а также виртуальный осциллограф фирмы Agilent) с отдельной регулировкой чувствительности в диапазоне от 10 мкВ/дел (mV/Div) до 5 кВ/дел (kV/Div) и регулировкой смещения по вертикали (Y POS). При этом каждый канал имеет два вывода обозначенные как «+» и «-». Подавая сигнал на один из выводов, другой целесообразно заземлять.

Выбор режима по входу осуществляется нажатием кнопок AC, 0, DC. Режим AC предназначен для наблюдения только сигналов переменного тока (его еще называют режимом «закрытого входа», поскольку в этом режиме на

входе усилителя включается разделительный конденсатор, не пропускающий постоянную составляющую). В режиме 0 входной зажим замыкается на землю. В режиме DC (включен по умолчанию) можно проводить осциллографические измерения как постоянного, так и переменного тока. Этот режим еще называют режимом «открытого входа», поскольку входной сигнал поступает на вход вертикального усилителя непосредственно. С правой стороны от кнопки DC расположен входной зажим.

Режим развертки выбирается кнопками Y/T, B/A, A/B. В режиме Y/T (обычный режим, включен по умолчанию) реализуются следующие режимы развертки: по вертикали – напряжение сигнала, по горизонтали – время; в режиме B/A: по вертикали – сигнал канала B, по горизонтали – сигнал канала A; в режиме A/B: по вертикали – сигнал канала A, по горизонтали – сигнал канала B.

Рис. 4. Осциллограф



В режиме развертки Y/T длительность развертки (TIME BASE) может быть задана в диапазоне от 0,1 нс/дел (ns/div) до 1 с/дел (s/div) с возможностью установки смещения в тех же единицах по горизонтали, т.е. по оси X (X POS).

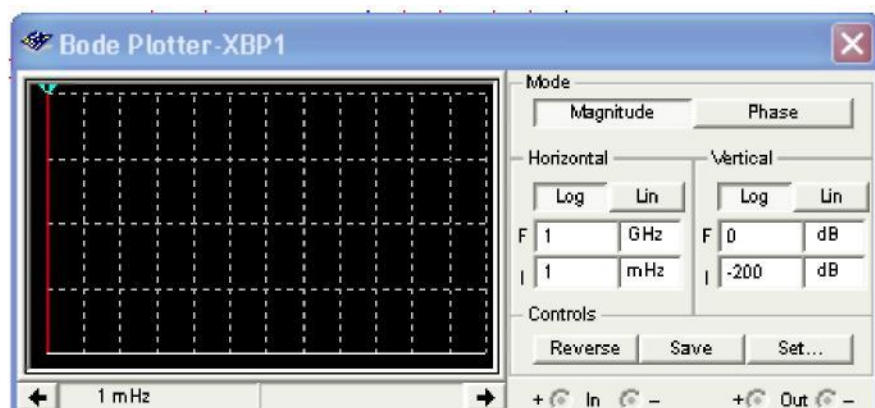
В режиме Y/T предусмотрен также ждущий режим (TRIGGER) с запуском развертки (EDGE) по переднему или заднему фронту запускающего сигнала (выбирается нажатием соответствующих кнопок) при регулируемом уровне (LEVEL) запуска, а также в режиме AUTO (от канала A или B), от канала A, от канала B или от внешнего источника (EXT), подключаемого к зажиму в блоке управления TRIGGER. Названные режимы запуска развертки выбираются кнопками AUTO, A, B, EXT.

Также пользователю доступна прокрутка изображения по горизонтали и его сканирования с помощью вертикальных визирных линий (синего и красного цвета), которые за треугольные ушки (они обозначены также цифрами 1 и 2) могут быть курсором установлены в любое место экрана. При этом в индикаторных окошках под экраном приводятся результаты измерения напряжения, временных интервалов и их приращений (между визирными линиями).

Изображение можно инвертировать нажатием кнопки REVERSE, а также можно записать данные в текстовый файл с помощью кнопки SAVE.

**Измеритель АЧХ и ФЧХ (Bode Plotter).** Лицевая панель измерителя АЧХ-ФЧХ показана на рис. 5. Измеритель предназначен для анализа амплитудно-частотных (при нажатой кнопке MAGNITUDE, включена по умолчанию) и фазо-частотных (при нажатой кнопке PHASE) характеристик при логарифмической (кнопка LOG, включена по умолчанию) или линейной (кнопка LIN) шкале по осям Y (VERTICAL) и X (HORIZONTAL). Настройка измерителя заключается в выборе пределов измерения коэффициента передачи и вариации частоты с помощью кнопок в окошках  $F$  – максимальное и  $I$  – минимальное значение. Значение частоты и соответствующее ей значение коэффициента передачи или фазы индицируются в окошках в правом нижнем углу измерителя. Значения указанных величин в отдельных точках АЧХ или ФЧХ можно получить с помощью вертикальной визирной линии, находящейся в исходном состоянии в начале координат и перемещаемой по графику мышью или кнопками  $\leftarrow$ ,  $\rightarrow$ . Результаты измерения можно записать также в текстовый файл. Для этого необходимо нажать кнопку SAVE и в диалоговом окне указать имя файла (по умолчанию предлагается имя схемного файла). В полученном таким образом текстовом файле «\*.scr» АЧХ и ФЧХ представляются в табличном виде.

Рис. 5. Измеритель АЧХ и ФЧХ



Подключение прибора к исследуемой схеме осуществляется с помощью зажимов IN (вход) и OUT (выход). Левые клеммы зажимов подключаются соответственно к входу и выходу исследуемого устройства, а правые – к общей шине (земля). Ко входу устройства необходимо подключить

функциональный генератор или другой источник переменного напряжения, при этом каких-либо настроек в этих устройствах не требуется.

#### 4. Анализ схем

Концепция электронной лаборатории с виртуальными измерительными приборами, заложенная в систему моделирования EWB, существенно облегчает проведение самого сложного этапа – расчета процессов, протекающих в радиоэлектронном устройстве. После составления схемы и подключения к схеме измерительных приборов для начала анализа цепи достаточно нажать кнопку *Activate/Stop*, находящуюся в левом углу панели инструментов. Рассчитанные значения токов, напряжений или сопротивлений показываются на экранах измерительных приборов. Аналогичный порядок работы имеет место в реальной лаборатории с реальными измерительными приборами. Процесс моделирования можно временно приостановить, нажав кнопку *Pause*. Для запуска, прекращения или временного приостановления процесса моделирования можно также воспользоваться соответствующими командами пункта меню или «горячими» клавишами (*F5* запуск и остановка вычислений, *F6* - пауза).

На втором этапе моделирования можно изменить параметры элементов, удалить или добавить радиоэлементы, подключить приборы к другим контрольным точкам схемы и т.п. После таких изменений, как правило, требуется снова активизировать цепь, нажимая кнопку *Simulate/Stop*. При использовании переменных резисторов, конденсаторов или катушек изменение процессов в цепи можно наблюдать, как правило, не прекращая моделирование. Однако при этом увеличивается погрешность получаемых результатов. Поэтому для получения надежных результатов расчет рекомендуется повторить при фиксированных параметрах.

В зависимости от типа подключенного прибора программа Multisim автоматически настраивается на выполнение следующих основных видов анализа:

- *DC Operating Point* – расчет режима по постоянному току, при включении мультиметра, амперметров и вольтметров для измерения постоянных токов и напряжений;
- *AC Analysis* – расчет частотных характеристик, при включении измерителя АЧХ и ФЧХ, а также мультиметра, амперметров и вольтметров для измерения гармонических токов и напряжений;
- *Transient* – расчет переходных процессов, при использовании осциллографа.

В программе Multisim предусмотрен также другой порядок анализа схемы – выбор режимов анализа с помощью подпункта меню *Simulate/Analysis*. Указанные выше режимы анализа электрической цепи можно получить, выбирая соответствующие команды подпункта *Analysis*. При выборе того или иного метода анализа всплывает диалоговое окно с

несколькими закладками, в котором установить параметры моделирования (количество закладок, их комбинация, а также содержание некоторых из них может варьироваться для различных методов анализа). Для каждого вида анализа будет предложен свой набор параметров для изменения. Так, например, в закладке **Analyses Parameters** будут предложены параметры характерные только для данного вида анализа. Например для AC Analysis будет предложено выбрать диапазон и изменения частоты, количество значений частоты для которых будет вычислена амплитудно- и фазо-частотная характеристика, характер выбора частоты расчёта (линейный выбор точки, через декаду, через октаву), а также параметры графиков, на которых будут выведены результаты (логарифмическая или линейная шкала). Некоторые закладка будут одинаковыми для всех видов анализа. Например, **Output** (позволяет выбрать узлы схемы для которых будут отображаться результаты расчёта и переменные, которые будут отображаться) и **Analysis Options**. Последняя закладка позволяет менять некоторые параметры численной схемы, используемой для моделирования предложенной пользователем радиофизической схемы. После нажатия на данной закладке кнопки **Customize** появляется диалоговое окно, содержащее пять основных закладок. Опишем некоторые параметры, которые могут пригодиться при выполнении заданий данного пособия.

**Закладка «Global»** - настройки общего характера, задаются с помощью следующих параметров:

- ABSTOL - абсолютная ошибка расчета токов;
- GMIN - минимальная проводимость ветви цепи (проводимость ветви, меньшая GMIN, считается равной нулю);
- RELTOL - допустимая относительная ошибка расчета напряжений и токов;
- VNTOL - допустимая ошибка расчета напряжений в режиме Transient (анализ несходных процессов);
- CHGTOL - допустимая ошибка расчета зарядов;
- RAMPTIME - начальная точка отсчета времени при анализе переходных процессов
- CONVSTEP - относительный размер шага итерации при расчете режима по постоянному току;
- CONVLIMIT – включение или выключение дополнительных средств для обеспечения сходимости итерационного процесса;
- RSHUNT – допустимое сопротивление утечки для всех узлов относительно общей шины (заземления).
- ACCT – разрешение на вывод статистических сообщений о процессе моделирования.

**Закладка «DC»** – настройка для расчета режима по постоянному току (статический режим):

- ITL1 – максимальное количество итераций приближенных

расчетов;

- **GMINSTEPS** – размер приращения проводимости в процентах от **GMIN** (используется при слабой сходимости итерационного процесса);

Кнопка **Reset Defaults** предназначена для установки по умолчанию параметров и используется в том случае, если после редактирования необходимо вернуться к исходным данным.

**Закладка «Transient»** – настройка параметров режима анализа переходных процессов:

- **ITL4** – максимальное количество итераций за время анализа переходных процессов;

- **MAXORD** – максимальный порядок (от 2 до 6) метода интегрирования дифференциального уравнения;

- **TRTOL** – допуск на погрешность вычисления переменной;

- **METHOD** – метод приближенного интегрирования дифференциального уравнения: **TRAPEZOIDAL** – метод трапеций, **GEAR** – метод Гира;

После установки всех необходимых значений параметров нажмите кнопку **Simulate** для активации процесса вычисления.

В электронной системе моделирования **Multisim** по умолчанию установлен

достаточно большой шаг численного интегрирования. Для повышения точности и корректности результатов анализа переходных процессов, особенно в узкополосных цепях, в цепях с нелинейными элементами и в других сложных цепях, рекомендуется в пункте меню **Transient** установить следующие значения параметров: **ITL4 = 100...1000** и **TRTOL = 1 ... 0.1**

Кроме указанных трех основных видов анализа, с помощью подпункта **Analysis** можно дополнительно провести другие виды анализа: спектральный анализ (**Fourier**), расчет чувствительности и разброса характеристик схемы при изменении параметров компонентов (**Monte Carlo**), анализ спектра внутренних шумов (**Noise**), расчет нелинейных искажений (**Distortion**), анализ влияния вариаций параметра какого либо элемента схемы (**Parameter sweep**), анализ влияния изменения температуры на характеристики устройства (**Temperature sweep**), расчет нулей и полюсов передаточной характеристики моделируемой цепи (**Pole-Zero**), расчет передаточной функции (**Transfer Function**) и расчет чувствительности и разброса характеристик схемы при изменении параметров компонентов (**Sensitivity, Worst Case**).

Расчет чувствительности и разброса характеристик схемы при изменении параметров компонентов может быть важен из-за того, что все радиоэлементы изготавливаются на предприятиях с некоторым разбросом параметров. В бытовой аппаратуре разброс параметров элементов достигает 20%. В аппаратуре специального назначения, в ответственных узлах радиоэлектронных устройств разброс параметров, как правило, не должен

превышать 10%. Отдельные радиокомпоненты изготавливаются с разбросом параметров, равным 5%, 2%, 1% и менее. Например, в источниках питания ЭВМ для получения высокостабильного напряжения могут использоваться резисторы с разбросом параметров, равным 0.5%. Для расчета нестабильности характеристик устройства и их чувствительности к изменениям параметров компонентов при заданном разбросе параметров элементов используется пункт меню Analysis > Monte Carlo. При анализе нестабильности характеристик, как правило, задается гауссово распределение случайного разброса параметров радиоэлементов. Анализ чувствительности и разброса характеристик схемы при изменении параметров компонентов требуется для оценки работоспособности устройств при модернизации, ремонте и при серийном производстве.

*Составлено на основе :  
Multisim™  
Руководство пользователя  
National Instruments Corporation*