

*Томский государственный университет
систем управления и радиоэлектроники*

А.С. Красько

ВЫЧИСЛИТЕЛЬНАЯ ПРАКТИКА

Учебно-методическое пособие



ТОМСК – 2006

Федеральное агентство по образованию Российской Федерации

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

Кафедра радиоэлектроники и защиты информации (РЗИ)

А.С. Красько

ВЫЧИСЛИТЕЛЬНАЯ ПРАКТИКА

Учебно - методическое пособие

ТУСУР – 2006

Красько А.С. Вычислительная практика: Учебно-методическое пособие. – Томск: Издательство ТУСУР, 2006. – 60 с.

В учебном пособии приведен материал, используемый на вычислительной практике. Программой практики предполагается знакомство студентов – первокурсников с современными математическими и схемотехническими пакетами на примере пакетов MathLab и Electronics WorkBench.

Предполагается, что студенты, приступающие к выполнению программы практики, владеют основами линейной алгебры и имеют опыт программирования в среде Pascal.

Навыки, приобретенные студентами во время вычислительной практики, будут полезны при изучении курсов основ теории электрических цепей, основ компьютерного проектирования и моделирования электронных схем, а так же при выполнении курсовых проектов в рамках курсов «Схемотехника АЭУ», УПОС и других инженерных дисциплин.

Для студентов, обучающихся по направлению подготовки 210300 – «Радиотехника».

© Красько А.С., 2006

© Томский государственный
университет систем управления
и радиоэлектроники, 2006

ОГЛАВЛЕНИЕ

№ раздела	Наименование	Стр.
1	Задачи и содержание вычислительной практики	4
2	Методические указания по прохождению вычислительной практики	4
3	Краткие теоретические сведения	5
	Список рекомендуемой литературы	10
	ПРИЛОЖЕНИЕ А. Краткое описание пакета Electronics WorkBench	11
	ПРИЛОЖЕНИЕ Б. MatLab. Краткое описание	38
	ПРИЛОЖЕНИЕ В. Индивидуальные задания	58
	ПРИЛОЖЕНИЕ Г. Бланк задания и требования к отчету	62
	ПРИЛОЖЕНИЕ Д. Примеры m- файлов	65

1 ЗАДАЧИ И СОДЕРЖАНИЕ ВЫЧИСЛИТЕЛЬНОЙ ПРАКТИКИ

Руководствуясь целями и задачами подготовки специалистов направления 210300 «Радиотехника», выделим основные задачи вычислительной практики:

- показать связь задач прикладной математики и задач расчета характеристик электрических цепей;
- получить практические навыки работы со специализированными математическими и схемотехническими пакетами;
- закрепить навыки создания электронных отчетов о выполненной работе.

В процессе вычислительной практики студенты в среде MatLab создают и отлаживают программу расчета характеристик электрических цепей, рассчитывают характеристики конкретной цепи, проверяют правильность полученных результатов с помощью простых инженерных соотношений и путем виртуального эксперимента в среде Electronics WorkBench (EWB).

По результатам работы предполагается подготовить в среде Office-2003 электронный вариант отчета, оформленный согласно требований стандарта ТУСУР.

2 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ПРОХОЖДЕНИЮ ВЫЧИСЛИТЕЛЬНОЙ ПРАКТИКИ

Задание практики выполняется в средах MatLab (версии 6.5 и выше) и Electronics Workbench (версии 5.12 и выше). Теоретические сведения, необходимые для выполнения лабораторных работ, содержатся в [1], в данном пособии приведен лишь минимально необходимый их объем. Сведения о пакетах MatLab и Electronics Workbench можно почерпнуть из, соответственно, [2] и [3], краткие сведения об этом пакете содержатся в Приложениях А и Б данного пособия.

Студент должен:

- получить индивидуальное задание у преподавателя, ведущего практику (варианты заданий и необходимые расчетные соотношения приведены в Приложении В);
- В системе **MatLab** создать и отладить программу расчета амлитудно-частотной (АЧХ) и фазо-частотной (ФЧХ) характеристик электрических цепей, реализующую обобщенный метод узловых потенциалов (ОМУП). На первом этапе выполнения задания рекомендуется в качестве схемы использовать интегральную цепочку (рис. 1.1). В главной программе использовать конструкцию **menu** для организации удобного для пользователя интерфейса. Получить, используя разработанную программу, АЧХ и ФЧХ интегральной цепи и схемы индивидуального задания.

- Используя пакет **Electronic WorkBench (EWB)**, проверить с помощью измерителя частотных характеристик (**Bode Plotter**) соответствие полученных АЧХ и ФЧХ расчетным, сделать соответствующие выводы.
- Написать отчет. Отчет представить в электронном виде на CD (но не на дискетах!!!).

Отчет оформляется согласно требованиям настоящего пособия (приложение Г), согласно общим правилам, принятым в настоящее время в ТУСУРе.

Отчет должен содержать:

- формулировку задачи;
- описание алгоритма и программы;
- таблицы и графики результатов с достаточным для понимания количеством комментариев;
- выводы по существу работы.

3 КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Чисто с математической точки зрения, задача сводится к формированию и решению систем линейных алгебраических уравнений, поэтому достаточно сведений о теории электрических цепей в объеме курса физики средней школы.

Расчет частотных характеристик электрических цепей в силу ряда причин предпочтительнее проводить с помощью метода узловых потенциалов (МУП). При использовании МУП исследуемая схема замещается **матрицей эквивалентных проводимостей**, отображающей как конфигурацию, так и свойства некоторой линейной схемы, аппроксимирующей реальную схему, элементы которой представляются в виде проводимостей.

При анализе линейных электрических цепей в частотной области с помощью МУП для нахождения напряжений в узлах решают уравнение вида:

$$\mathbf{I} = \mathbf{Y} \cdot \mathbf{U} \quad (1)$$

где \mathbf{I} – вектор задающих токов (вектор свободных членов);

\mathbf{Y} – матрица эквивалентных проводимостей (матрица коэффициентов);

\mathbf{U} – вектор узловых напряжений (вектор неизвестных).

Вектора \mathbf{I} и \mathbf{U} – имеют размерность $(1 \times n)$, где n – число узлов схемы без общего (заземленного) узла

Матрица эквивалентных проводимостей схемы $[\mathbf{Y}]$ имеет размерность $(n \times n)$, где n – число узлов схемы без общего (заземленного) узла и составляется непосредственно по схеме, используя следующий алгоритм:

♦ диагональные элементы матрицы $[\mathbf{Y}]$ y_{kk} вычисляются как сумма проводимостей ветвей схемы, подходящих к рассматриваемому узлу k , $k=1, \dots, n$;

◆ недиагональные элементы матрицы $[Y]$ y_{kl} вычисляются как сумма проводимостей ветвей схемы, соединяющих узлы k и l , ($k, l=1, \dots, n; k \neq l$), взятая с обратным знаком.

Проводимости ветвей вычисляются следующим образом:

$$g=1/R \text{ (См)},$$

если ветвь содержит в качестве элемента резистор с сопротивлением $R(\text{Ом})$;

$$y_C=j\omega C \text{ (См)}$$

если ветвь содержит в качестве элемента конденсатор с емкостью $C(\text{Ф})$, y_C носит комплексный характер, здесь $\omega=2\pi f$, где f – текущая частота, Гц.;

$$y_L=1/j\omega L \text{ (См)}$$

если ветвь содержит в качестве элемента катушку индуктивности с индуктивностью $L(\text{Гн})$, y_L носит комплексный характер, здесь $\omega=2\pi f$, где f – текущая частота, Гц.;

Информация об независимом источнике тока J заносится в вектор $[I]$ следующим образом:

◆ со знаком плюс в элемент вектора $[I]$, соответствующий номеру узла, куда втекает ток источника;

◆ со знаком минус в элемент вектора $[I]$, соответствующий противоположному случаю.

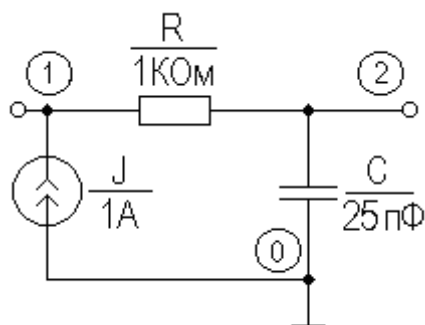


Рисунок 3.1 – Схема интегрирующей цепи

Рассмотрим реализацию этого алгоритма на примере схемы рисунка 3.1.

В этом случае на частоте $\omega=1$ система (1) приобретет вид:

$$\begin{matrix} & 1 & & 2 \\ \begin{matrix} 1 \\ 2 \end{matrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} & = & \frac{1}{2} \begin{bmatrix} 0.001 & -0.001 \\ -0.001 & 0.001 + j25 \cdot 10^{-12} \end{bmatrix} \cdot \begin{bmatrix} U_1 \\ U_2 \end{bmatrix}. \end{matrix}$$

Алгоритмы формирования и решения уравнений (1). Для формирования уравнений узловым методом необходима информация о количестве узлов схемы, для контроля правильности ввода исходных данных желательна информация и о количестве ветвей схемы. Эта информация должна определяться автоматически каждый раз, как только в программу вводятся данные о новой схеме.

Примем следующую кодировку двухполюсных элементов схемы:

признак, номинал, узлы подключения,

т.е. двухполюсный элемент будет описываться четырьмя элементами.

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

Условимся о следующей кодировке признака элемента:

Проводимость – **G**

Емкость - **C**

Индуктивность - **L**

Источник тока - **I**

В этом случае информация, например, об конденсаторе **C** схемы рисунка 1, представится следующим образом (через пробелы):

'C' 25e12 2 0

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

G – Ом (1/Сим), **C** - Ф, **L** - Генри, **I** - А.

Для схемы рисунка 1 массив данных будет выглядеть следующим образом:

dd=['I' 1 1 0 'G' 1e3 1 2 'C' 25e-9 2 0];

Первая файл-функция должна определить порядок системы уравнений ($n=2$ для рассматриваемого примера) и количество ветвей ($b=3$ для этого примера). В Приложении Д приведен в качестве примера листинг файл – функции (процедуры) **ytop**, которая реализует данную задачу.

Используя составленный файл данных и найденное значение **n**, формируют матрицу коэффициентов [**Y**] и вектор свободных членов [**I**].

Алгоритм файл-функции формирования достаточно прозрачен и состоит из следующих операций:

- ▲ предварительного обнуления матрицы [**Y**] и вектора [**I**];
- ▲ организации цикла по элементам с определением их типа, выделением номинала и узлов подключения;
- ▲ занесения информации об элементе в матрицу [**Y**] (по необходимости, и в вектор [**I**]) согласно вышеописанному алгоритму.

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.

В Приложении Д приведен в качестве примера листинг файл – функции (процедуры) **ymatr**, которая реализует данную задачу.

Результатом решения схемных уравнений МУПр (1) являются узловые потенциалы.

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:

$$\dot{K} = \dot{U}_{\hat{a}\hat{o}} / \dot{U}_{\hat{a}\hat{i}} = |K| \exp(j\varphi);,$$

где φ - фазовый сдвиг между входным и выходным сигналами.

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

Расчет АЧХ и ФЧХ можно проводить по следующему алгоритму:

▲ вводится информация об элементах схемы и номера узлов, между которыми ищется коэффициент передачи;

▲ определяются количество узлов и ветвей второй группы;

▲ организуется цикл по частоте с заданным шагом;

▲ 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;

▲ накапливается массив значений модуля, фазы, точек частотного диапазона и строятся АЧХ и ФЧХ.

В Приложении Д приведен в качестве примера листинг script – файла (файла – сценария, или управляющей программы) **ymein**, которая реализует данную задачу.

Рассмотрим в качестве примера расчет АЧХ и ФЧХ интегрирующей цепи (рисунок 1). 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:

$$f_{\hat{a}} = \frac{1}{2\pi\tau},$$

где τ - постоянная времени цепи, $\tau = RC$.

Подставив в выражение для f_g значения элементов, указанные на рисунке 3.2, получим значение $f_a \approx 6,4 \cdot 10^6 \text{ \AA} \ddot{\text{o}}$. Величина фазового сдвига на граничной частоте составляет для этой цепи -45° .

Результаты расчета интегрирующей цепи программой, составленной по данным рекомендациям, приведены на рисунке 2.1

Как видно из результатов расчета, значение f_g 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.

Требования к программе. Программа создается в среде MatLab с использованием всех доступных средств пакета: функций, языка программирования, меню, графики и т.д.

Оформить программы получения информации о схеме и формирования системы схемных уравнений в виде *файл – функций (процедур)*. Написать и отладить *script – файл (файл – сценарий, или управляющую программу)* для расчета АЧХ и ФЧХ электрических цепей с использованием вышеописанных процедур, организовать графический вывод результатов анализа, протестировать программу путем анализа простейших цепей. Схема индивидуального задания согласовывается с преподавателем, ведущим практику. Примеры файл – функций и script – файл даны в Приложении Д

СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ

1. Красько А.С. Схемотехника аналоговых электронных устройств. Учебное пособие. – Томск: В-Спектр, 2006. – 180 с.
2. Потемкин В.Г. Система MATLAB. Справочное пособие. - М.: Диалог МИФИ, 1997. – 350 с.
3. Карлащук В.И. Электронная лаборатория на IBM PC. Программа Electronics Workbench и ее применение.–М.: “Солон-Р”, 1999. – 506 с.
4. Любая другая литература по указанной тематике

ПРИЛОЖЕНИЕ А

1 КРАТКОЕ ОПИСАНИЕ ПАКЕТА ELECTRONICS WORKBENCH

1.1 Внешний интерфейс пользователя Electronics Workbench

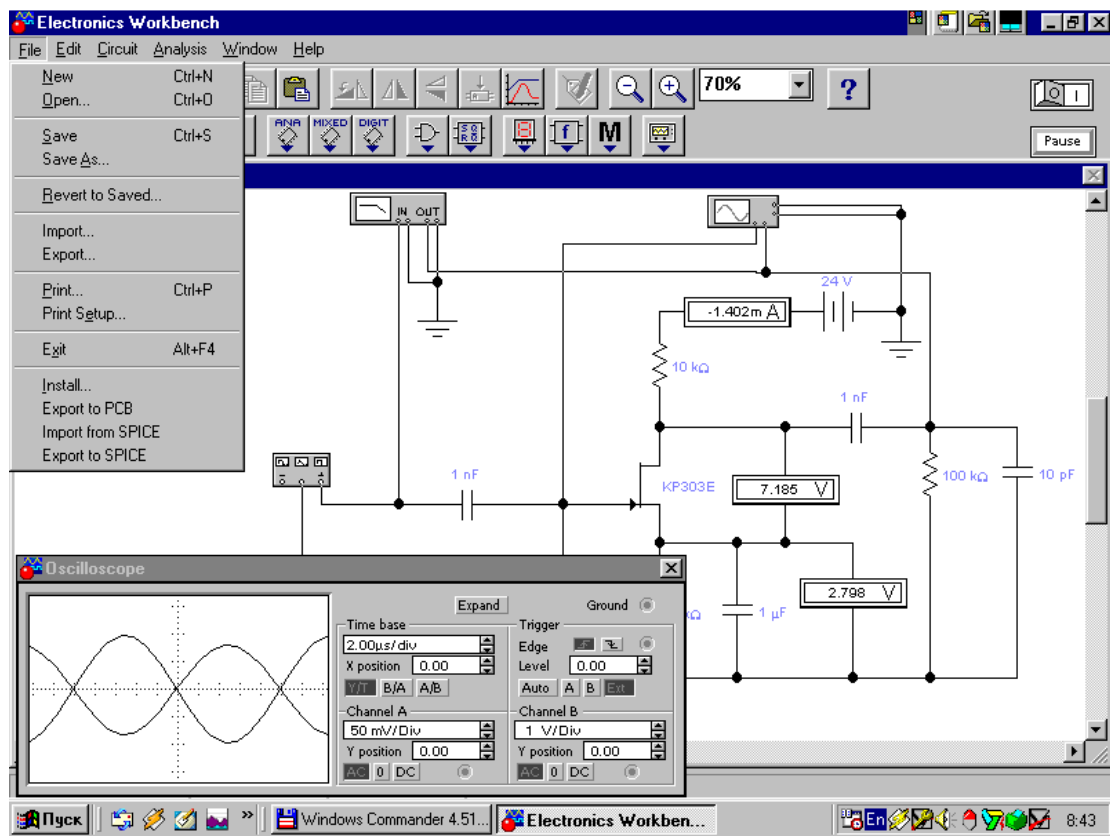


Рисунок 1 – Внешний вид экрана компьютера при работе с программой EWB

Приложение Electronics Workbench v.5.12 (EWB) представляет собой средство программной разработки и имитации электрических цепей.

Интерфейс пользователя состоит из полосы меню, панели инструментов и рабочей области.

Полоса меню состоит из следующих компонент: меню работы с файлами (File), меню редактирования (Edit), меню работы с цепями (Circuit), меню анализа

схем (Analysis), меню работы с окнами (Window), меню работы с файлами справоч (Help).

Панель инструментов состоит из “быстрых кнопок”, имеющих аналоги в меню, кнопок запуска и приостановки схем, набора радиоэлектронных аналоговых и цифровых деталей, индикаторов, элементов управления и инструментов.

Меню File

Меню File позволяет осуществить операции работы с файлами.

<u>N</u>ew	Ctrl+N
<u>O</u>pen...	Ctrl+O
<u>S</u>ave	Ctrl+S
Sa<u>v</u>e <u>A</u>s...	
<u>R</u>evert to Saved...	
I<u>m</u>port...	
E<u>x</u>port...	
<u>P</u>rint...	Ctrl+P
P<u>r</u>int <u>S</u>etup...	
P<u>r</u>ogram <u>O</u>ptions...	
E<u>x</u>it	Alt+F4
<u>I</u>nstall...	

Рисунок 2 – внешний вид меню File

File/New. Операцию также можно вызвать одновременным нажатием клавиш **CTRL+N**.

Данная операция предназначена для закрытия текущей схемы и создания новой. При этом создается безымянное окно, которое может использоваться для создания схемы. Если перед этим вы проделали какие-либо изменения текущей схемы, вам будет предложено сохранить текущую схему перед ее закрытием. При запуске Electronics workbench операция выполняется автоматически. По умолчанию схема именуется как Default.ewb.

File/Open. Операцию также можно вызвать одновременным нажатием клавиш **CTRL+O**.

Операция предназначена для открытия уже существующего файла схемы. Отображает стандартное диалоговое окно открытия файла, в котором необходимо выбрать диск и каталог, содержащий файл схемы, который вы хотите открыть. Открывать можно только файлы с расширениями .ca, .ca3, .cd3, .ca4 и .Ewb.

File/Save. Операцию также можно вызвать одновременным нажатием клавиш **CTRL+S**.

Сохраняет текущий файл схемы. Отображается стандартное диалоговое окно сохранения файла, в котором необходимо выбрать диск и каталог, где вы хотите сохранить схему и название файла. Расширения .Ewb добавляются к имени файла автоматически. Например, схема с именем Music, будет сохранена как Music.ewb.

File/Save as. Команда аналогична операции ***Save***, но сохраняет текущую схему с новым именем файла, оставляя первоначальную схему неизменной.

Используйте эту команду, чтобы безопасно экспериментировать на копии схемы, без изменения оригинала.

File/Revert to Saved (Revert). Эта команда восстанавливает схему к виду, который она имела в момент последнего сохранения.

File/Import. Команда преобразует нестандартные файлы схем (расширение .net или .cir) и преобразовывает их к стандартному виду Electronics Workbench.

File/Export. Сохраняет файл схемы с одним из следующих расширений: .net, .scr, .cmp, .cir, .plc.

File/Print. Операцию также можно вызвать одновременным нажатием клавиш **CTRL+P**.

Команда предназначена для полной или частичной распечатки схемы и/или приборов. Для выполнения операции необходимо выбрать элементы, которые будут напечатаны, в порядке, в котором вы хотите их напечатать.

File/Print Setup (Windows). Эта операция предназначена для настройки принтера. Отображает стандартное диалоговое окно Print Setup, из которого Вы можете выбрать установленный принтер и определять ориентацию изображения, бумажный размер, бумажный источник и другие параметры. Для схем, которые по ширине больше чем по высоте, используйте альбомную ориентацию. Если схема слишком велика для печати на одном листе, печать будет автоматически производиться на нескольких страницах.

File/Exit. Операцию также можно вызвать одновременным нажатием клавиш **ALT+F4**.

Операция предназначена для завершения работы с пакетом Electronics Workbench. Если Вы не сохранили изменения в схеме, то будет сделан запрос на сохранение.

File/Install (Windows). Операция предназначена для установки добавочных компонент Electronics workbench. Для ее выполнения будет запрошен диск, содержащий дополнительные компоненты.

Меню Edit

C<u>u</u>t	Ctrl+X
C<u>o</u>py	Ctrl+C
P<u>a</u>ste	Ctrl+V
D<u>e</u>lete	Del
S<u>e</u>lect <u>A</u>ll	Ctrl+A
<hr/>	
C<u>o</u>py as B<u>i</u>tmap	
S<u>h</u>ow C<u>l</u>ipboard	

Меню Edit позволяет осуществить операции редактирования.

Рисунок 3 – Внешний вид меню Edit

Edit/Cut. Операцию также можно вызвать одновременным нажатием клавиш **CTRL+X**.

Команда используется для удаления выбранных компонент, схем или текста. При этом выбранное помещается в буфер обмена, откуда его можно вставлять в нужное место. Команда не сработает, если выбор включает в себя инструментальные пиктограммы.

Edit/Copy. Операцию также можно вызвать одновременным нажатием клавиш **CTRL+C**.

Команда предназначена для копирования выбранных компонент, схемы или текста. Копия помещается в буфер обмена. Затем вы можете использовать команду Paste, чтобы вставить копию в нужном месте. Операция также не выполнится, если выбор включает инструментальные пиктограммы.

Edit/Paste. Операцию также можно вызвать одновременным нажатием клавиш **CTRL+V**.

Команда помещает содержание Буфера обмена в активное окно (содержание остается в Буфере обмена). Для успешного выполнения операции Буфер должен содержать компоненты Electronics Workbench или текст.

Содержимое Буфера обмена может вставляться только в окна, способные содержать подобную информацию. Например, Вы не можете вставить компонент электрической схемы в окно описания.

Edit/Delete. Операцию также можно вызвать нажатием клавиши **DEL**.

Эта команда полностью удаляет выбранные компоненты или текст. Используйте команду Delete с осторожностью. Удаленная информация не может быть восстановлена.

Edit/Select All. Команда выбирает все элементы в активном окне (окно схемы, окно подсхемы или окно описания). Если прибор - часть выбора, команды Edit/Copy и Edit/Paste становятся недоступными. Для того, чтобы выбрать все, кроме нескольких элементов, используйте команду Select All, и затем снимите выделение с лишних элементов, нажимая CTRL с левой кнопкой мыши.

Edit/Copy as Bitmap. Команда предназначена для копирования растрового изображения элементов в Буфер обмена. Вы можете использовать эти изображения в текстовых процессорах или программах обработки изображений.

Чтобы скопировать растровое изображение элементов необходимо:

- а) выбрать Edit/Copy as Bitmap (курсор изменится на crosshair),
- б) нажать и удерживать кнопку мыши перемещая курсор, чтобы сформировать прямоугольник, включающий необходимые для копирования элементы,
- в) отпустить кнопку мыши.

Edit/Show Clipboard. Команда отображает содержание Буфера обмена. Буфер обмена - временное место хранения для компонентов или текста, которые Вы хотите поместить позже в другом месте в схеме. Вы можете также использовать Буфер обмена, чтобы передать информацию от Electronics workbench к другой прикладной программе. Буфер обмена может содержать графику (компоненты или схемы) и текст. Если активное окно не может содержать тип информации, которая находится на Буфере обмена, или если

Буфер обмена пуст, команда Edit/Paste будет не доступна. Например, если Буфер обмена содержит компоненты, а текущим является окно описания, команда Paste будет недоступна. Чтобы закрыть Буфер обмена, дважды щелкните меню Control(Windows).

Меню Circuit

Меню Circuit позволяет осуществить операции работы с цепями.

<u>R</u>otate	Ctrl+R
Flip Horizontal	
Flip Vertical	
Component Properties...	
<hr/>	
Create Subcircuit...	Ctrl+B
Zoom In	Ctrl++
Zoom Out	Ctrl+-
<hr/>	
Schematic Options...	
Restrictions...	Ctrl+I

Рисунок 4 – Внешний вид меню Circuit

Circuit/Rotate. Операцию также можно вызвать одновременным нажатием клавиш **CTRL+R**

Команда позволяет вращать выбранные компоненты на 90 градусов по часовой стрелке. Текст, связанный с компонентом (метки, значения и информация о модели), может быть повторно установлен, но при выполнении команды не вращается. В случае необходимости, провода, приложенные к компоненту перенаправляются автоматически. Когда Вы вращаете амперметр и вольтметр, вращаются только их терминалы.

Circuit/Flip Vertical. Команда зеркально отражает выбранную схему по вертикали в окне схемы. Обратите внимание, любые провода, приложенные к зеркально отражаемому компоненту перенаправляются по мере необходимости. Текст, связанный с компонентом (метки, значения и информация о модели), может быть повторно установлен, но не отражается.

Circuit/Flip Horizontal. Команда зеркально отражает выбранную схему по горизонтали в окне схемы. Любые провода, приложенные к зеркально отражаемому компоненту перенаправляются по мере необходимости. Текст, связанный с компонентом (метки, значения и информация о модели), может быть повторно установлен, но не отражается.

Circuit/Component Properties. Команда предназначена для изменения свойств выбранного компонента. Также выводится при двойном нажатии на компоненте. При вызове с помощью всплывающего меню, после нажатия правой кнопкой мыши, назначаются заданные по умолчанию свойства для всех выбранных компонентов, впоследствии используемых в этой схеме. Это не воздействует на уже размещенные компоненты.

При выполнении команды открывается диалоговое окно *Circuit/Component Properties*, закладки которого зависят от типа выбранного компонента.

Возможны следующие типы закладок:

- Label,
- Value,
- Models,
- Schematic Options,
- Fault,
- Node,
- Display,
- Analysis Setup.

Закладка Label (Свойства компонента). Закладку также можно вызвать одновременным нажатием клавиш **CTRL+L**.

Используйте эту закладку, чтобы установить или заменить метку компонента и идентификатор (компоненты типа соединителей, заземлений, измерителей не имеют идентификаторов).

Если Вы вращаете или зеркально отражаете компонент, метка может быть установлена повторно. Если, в результате, провод проходит через метку, Вы можете сдвинуть метку направо, добавляя несколько пробелов перед меткой.

Чтобы вставить общую информацию в схему, введите текст в окно описания, доступное из меню Window.

Обратите внимание, идентификаторы назначаются системой уникально, идентифицируя компонент. Вы можете изменять их в случае необходимости, но они должны оставаться уникальными. Идентификаторы не могут быть удалены.

Value Tab (Свойства компонента). Закладку также можно вызвать одновременным нажатием клавиш **CTRL+U**. Поля на этой закладке различаются в зависимости от компонента.

Закладка Models (Свойства компонента).

Закладку также можно вызвать одновременным нажатием клавиш **CTRL+M**.

Используйте эту закладку, чтобы выбрать модель, используемую для компонента и для редактирования, добавления или удаления моделей или библиотек. Компоненты по умолчанию «идеальны», что для большинства схемотехнических моделирований может быть достаточным. Однако, если вы хотите увеличить точность результатов теста, используйте «реальную» модель.

Закладка Schematic Options (Свойства компонента). Закладка используется, чтобы установить цвет провода.

Закладка Fault (Свойства компонента). Закладку также можно вызвать одновременным нажатием клавиш **CTRL+F**.

Используйте эту закладку, чтобы назначить неисправность на терминал компонента.

Leakage - помещает значение сопротивления, определенное в смежных полях, параллельно с выбранными терминалами. Это заставляет ток течь мимо терминалов вместо того, чтобы пройти их.

Short - помещает очень низкое сопротивление между двумя терминалами, так что компонент не имеет никакого измеримого эффекта на схеме.

Open - помещает очень высокое сопротивление на терминале, как будто проводное соединение на терминал было разбито.

Закладка Node (Свойства компонента). Закладка используется для изменения свойств узла.

Node ID - назначенное системой имя узла.

Use as Testpoint - определяет, должен ли узел рассматриваться как тестовая точка.

Set Node Color - отменяет набор цветов для отдельных проводов.

Закладка Display (Свойства компонента). Закладка используется для отображения/скрытия тех или иных элементов Electronics Workbench.

Когда выбрано Use Schematic Options используются настройки параметров дисплея из закладки Show/Hide диалогового окна Circuit/Schematic Options.

Show labels, Show models, Show reference ID - когда не выбрано Use Schematic Options, используются параметры дисплея как они были определены.

Закладка Setup (Свойства компонента). Закладка используется для настройки параметров элементов, таких как рабочая температура.

Use global temperature - если выбрано, используется набор температур установленный в Analysis/Analysis Options. Если не выбрано, используются те температуры, которые были определены.

Set initial conditions - устанавливает начальные значения для компонента.

Некоторые компоненты отображают дополнительные параметры на этой закладке, для использования вместе с параметрами, описанными в техническом справочнике Electronics Workbench.

Circuit/Create Subcircuit. Команду также можно вызвать одновременным нажатием клавиш **CTRL+B**.

Команда объединяет выбранные элементы схемы в подсхему, в действительности создавая интегральную схему.

Подсхема может содержать так много компонентов, как требуется. Любая подача проводов к другим компонентам или соединителям в схеме станет терминалами на пиктограмме подсхемы.

Чтобы создать подсхему:

- а) Выберите элементы, которые нужно использовать для подсхемы.
- б) Выберите *Circuit/Create Subcircuit*, и завершите диалог, который появляется:

Copy from Circuit - помещает копию выбранных компонентов в подсхему. Первоначальные компоненты остаются, поскольку они находятся в окне схемы.

Move from Circuit - удаляет выбранные компоненты из схемы, так что они появляются только в подсхеме.

Replace in Circuit, помещает выбранные компоненты в подсхему и заменяет выбранные компоненты в схеме прямоугольником, помеченным именем подсхемы.

Выбранные компоненты появляются в новом окне, окне подсхемы. Имя новой подсхемы добавляется к списку доступных подсхем, который отображается, когда пиктограмма подсхемы перемещается из инструментальной панели Favorites. Подсхема доступна только для текущей схемы.

Circuit/Zoom. Команда отображает подменю выбора для увеличения или уменьшения размера дисплея окна схемы.

Circuit/Schematic Options. Команда предназначена для управления всем дисплеем схемы. Изменения относятся только к текущей схеме.

В окне команды выводится следующий набор закладок:

-Grid,

-Show/Hide,

-Display,

-Value.

Закладка Grid. Закладка управляет дисплеем и использованием сетки, лежащей в основе окна схемы. Использование сетки упрощает выравнивание элементов в схеме. Вы можете использовать сетку без ее отображения. Отображение сетки осуществляется на заднем плане окна схемы. Сетку удобно использовать при расстановке объектов.

Закладка Show/Hide. Закладка управляет дисплеем информации в окне схемы. Ее параметры полезно использовать, когда нужно скрыть объект.

Закладка Display. Закладка управляет шрифтом, используемым для меток и ссылок на идентификаторы.

Закладка Value. Закладка управляет шрифтом, используемым для значений и моделей.

Меню Analysis. Меню Analysis позволяет выполнить различные анализы. Внешний вид меню приведен на рисунке 5. Перед выполнением каждого из них пользователю будет предложено заполнить параметров анализа. Анализ будет выполнен только в том случае, когда это возможно для данной схемы.

<u>A</u>ctivate	Ctrl+G
R esume	F9
<u>S</u>top	Ctrl+T
A nalysis Options...	Ctrl+Y
D C Operating Point	
A C Freq <u>u</u> ency...	
T ransient...	
<u>F</u>ourier...	
<u>N</u>oise...	
<u>D</u>istortion...	
<u>P</u>arameter Sweep...	
T emperature Sweep...	
P ole-Z <u>e</u> ro...	
<u>T</u>ransfer Function...	
<u>S</u>ensitivity...	
<u>W</u>orst Case...	
<u>M</u>onte Carlo...	
D isplay Graphs	

Рисунок 5 – Внешний вид меню Analysis

Analysis/Activate. Команду также можно вызвать одновременным нажатием клавиш **CTRL+G**.

Команда активизирует схему (включает переключатель питания). Активизация схемы начинает последовательность математических операций, чтобы вычислить значения для тестовых точек в схеме.

Переключатель питания остается включенным, пока Вы не останавливаете или не приостанавливаете моделирование.

Analysis/Pause and Analysis/Resume. Команду также можно вызвать нажатием клавиши **F9**.

Команда временно прерывает или продолжает моделирование (управляется кнопкой Pause/Resume). Приостановка полезна, если вы хотите

рассмотреть форму волны (форму кривой, форму сигнала) или сделать изменения в инструментальных настройках. (Имитация простых схем может оказаться слишком быстрой для приостановки.)

Analysis/Stop. Команду также можно вызвать одновременным нажатием клавиш **CTRL+T**.

Команда вручную останавливает моделирование. Имеет тот же эффект что и щелчок переключателя питания.

Обратите внимание, что выключение энергии стирает данные и инструментальные следы и сбрасывает все значения к начальным.

Analysis/Analysis Options. Команду также можно вызвать одновременным нажатием клавиш **CTRL+Y**.

Electronics Workbench позволяет Вам управлять многими аспектами моделирования, типа сброса терпимости ошибки, выбор методов моделирования и просмотра результатов. Эффективность моделирования также зависит от параметров, которые Вы выбираете. Большинство параметров имеет значения по умолчанию.

Чтобы рассмотреть или изменить любые из параметров, выберите Analysis/Analysis Options.

Analysis/DC Operating Point. Команда выполняет анализ схемы по постоянному току.

Analysis/AC Frequency. Команда выполняет анализ__схемы по переменному току (частотный анализ).

Analysis/Transient. Команда выполняет анализ переходных процессов.

Analysis/Fourier. Команда выполняет Фурье - анализ.

Analysis/Noise. Команда выполняет шумовой анализ.

Analysis/Distortion. Команда выполняет анализ нелинейных и интермодуляционных искажений.

Analysis/Parameter Sweep. Команда выполняет анализ при вариации параметров элементов.

Analysis/Temperature Sweep. Команда выполняет температурный анализ.

Analysis/Pole-Zero. Команда выполняет нуль-поллюсной анализ.

Analysis/Transfer Function. Команда выполняет анализ передаточных функций.

Analysis/Sensitivity. Команда выполняет анализ чувствительности.

Analysis/Worst Case. Команда выполняет анализ значений параметров компонентов схемы при предельных отклонениях ее характеристик.

Analysis/Monte Carlo. Команда выполняет статистический анализ по методу Монте-Карло.

Analysis/Display Graph. Команда выводит графические результаты анализа.

Window Menu. Меню Window позволяет осуществить операции работы с окнами.

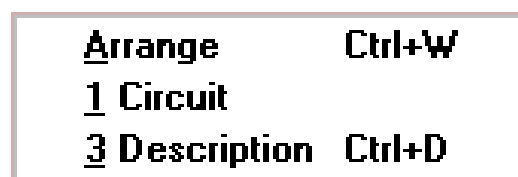


Рисунок 6 – Внешний вид меню Window

Window/Arrange. Команду также можно вызвать одновременным нажатием клавиш **CTRL+W**.

Команда аккуратно расставляет открытые окна.

Window/Circuit. Команда переносит окно схемы на передний план.

Window/Description. Команду также можно вызвать одновременным нажатием клавиш **CTRL+D**.

Команда открывает окно описания. (Если окно описания уже открыто, переносит его на передний план.) Вы можете напечатать комментарии или указания в окне описания, а также вставить текст из другой прикладной программы или описания схемы.

Меню Help. Меню Help предоставляет вызов файла-справки. Вызов справки также можно осуществить нажатием клавиши **F1**.

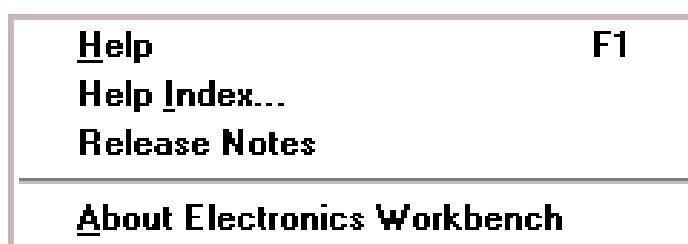


Рисунок 7 – Внешний вид меню Help

1.2 Порядок подготовки и анализа принципиальной электрической схемы

1. Запустите Electronics Workbench.

2. Подготовьте новый файл для работы. Для этого необходимо выполнить следующие операции из меню: File/New и File/Save as. При выполнении операции Save as будет необходимо указать имя файла и каталог, в котором будет храниться схема. Рекомендуется называть схему по фамилии исполнителя.

3. Перенесите необходимые элементы схемы на рабочую область Electronics Workbench. Для этого необходимо выбрать раздел на панели инструментов (Sources, Basic, Diodes, Transistors, Analog Ics, Mixed Ics, Digital Ics, Logic Gates, Digital, Indicators, Controls, Miscellaneous, Instruments), в котором находится нужный вам элемент, затем перенести его на рабочую область.

4. Соедините контакты элементов и расположите элементы в рабочей области для получения необходимой вам схемы. Для соединения двух контактов необходимо щелкнуть по одному из контактов основной кнопкой мыши и, не отпуская клавишу, довести курсор до второго контакта. В случае необходимости можно добавить дополнительные узлы (разветвления). Нажатием на элементе правой кнопкой мыши можно получить быстрый доступ к простейшим операциям над положением элемента, таким как вращение (rotate), разворот (flip), копирование/вырезание (copy/cut), вставка (paste).

5. Проставьте необходимые номиналы и свойства каждому элементу. Для этого нужно дважды щелкнуть мышью на элементе.

6. Когда схема собрана и готова к запуску, нажмите кнопку включения питания на панели инструментов. В случае серьезной ошибки в схеме (замыкание элемента питания накоротко, отсутствие нулевого потенциала в схеме) будет выдано предупреждение.

7. Произведите анализ схемы, используя инструменты индикации. Вывод терминала осуществляется двойным нажатием клавиши мыши на элементе. В случае надобности можно пользоваться кнопкой Pause.

8. При необходимости произведите доступные анализы в разделе меню Analysis.

2 МОДЕЛИРОВАНИЕ РАДИОЭЛЕКТРОННЫХ УСТРОЙСТВ ПРИ ПОМОЩИ ПРОГРАММНОГО КОМПЛЕКСА ELECTRONICS WORKBENCH

Подготовка к работе Electronics Workbench. Для работы с программным комплексом Electronics Workbench V.5.0С необходим IBM - совместимый персональный компьютер с процессором I486 (рекомендуется

Pentium) и операционной системой Windows 3.1 (рекомендуется Windows 95 или Windows 98).

Для начала работы с программным пакетом необходимо установить Electronics Workbench, если это не было сделано ранее. Затем при помощи «диспетчера файлов» (для Windows 3.1) или «проводника» (для Windows 95 или Windows 98) открыть рабочий каталог, в котором установлен пакет и запустить исполняемый файл WEWB32.EXE. Далее, при необходимости, можно произвести настройки интерфейса пользователя

Пример моделирования интегрирующей RC-цепи. Для начала разработки необходимо загрузить файл-схему в среду Electronics Workbench, если этот файл уже создан и находится на одном из накопителей компьютера. Это делается посредством выполнения команды меню File/Open, либо нажатием на соответствующей «горячей кнопке» на панели инструментов и дальнейшим выбором накопителя, каталога, и имени файла. Если же файл еще не создан - необходимо создать его посредством выполнения команды File/New и команды File/Save as. При выполнении первой команды будет создан новый файл-схема и в случае, если какая либо схема уже загружена в Electronics Workbench, пользователю будет предложено сохранить ее. Вторая команда предназначена для записи файла на накопитель и установки каталога и имени, под которым будет храниться данная схема.

Далее нужно нанести на рабочую область Electronics Workbench модели деталей необходимые для моделирования данной схемы. Это делается посредством нажатия левой кнопкой мыши на нужном наборе деталей, после чего будет выведено дополнительное окно включающее в себя детали набора, выбором соответствующей детали, при этом на кнопке с рисунком элемента нажимается левая кнопка мыши и элемент переносится на рабочую область

(кнопку мыши необходимо держать нажатой до выбора места расположения элемента. В данном случае

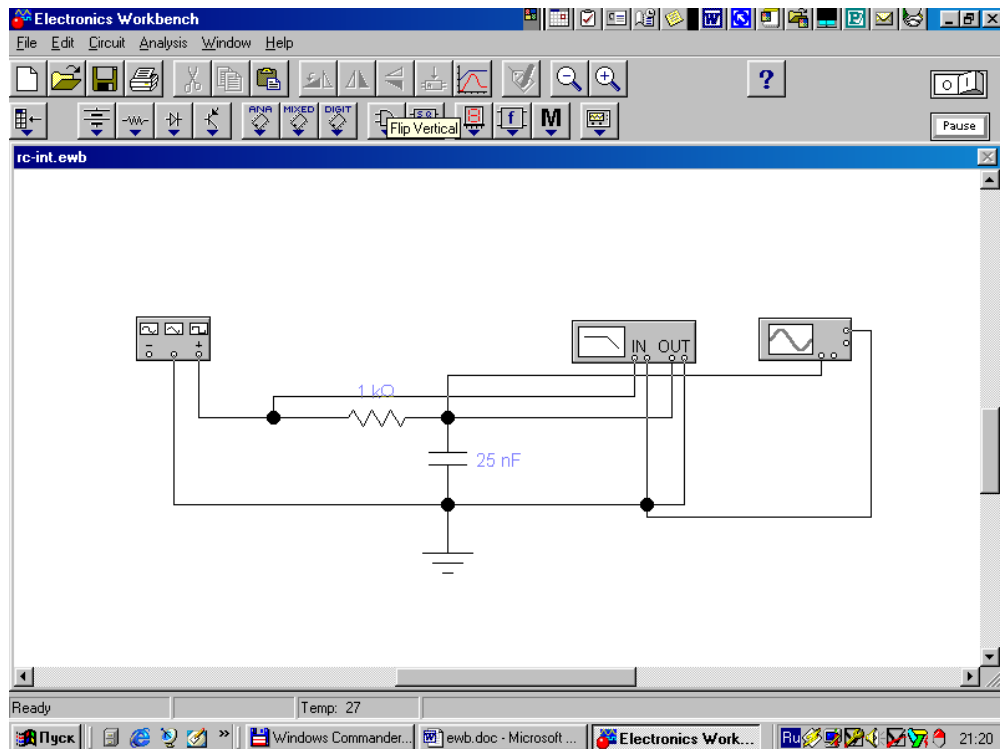


Рисунок 8 – RC-цепь в Electronics Workbench

необходимы: источник импульсов (Function Generator), резистор (Resistor), конденсатор (Capacitor), осциллограф (Oscilloscope) и заземление (Ground). Резистор и конденсатор находятся в наборе Basics, заземление - в наборе Sources, осциллограф и генератор импульсов - в наборе Instruments.

Каждый элемент имеет точки соединения, которые нужно соединить для получения нужной схемы. Это делается выбором контакта левой кнопкой мыши и переносом ее к другому контакту, при этом создается провод, соединяющий их. При необходимости на провод можно нанести узел (Connector в наборе Basics). Затем для наглядности можно перенести элементы в необходимые места рабочей области. Это действие также осуществляется нажатием на элементе левой кнопкой мыши и переносом при удержанной в нажатом состоянии кнопке. При этом соединительные провода будут перемещены автоматически. При

необходимости провода можно также перемещать. На рисунке 8 представлен вид интерфейса Electronics Workbench после сборки RC – цепи.

Когда схема создана и готова к работе для начала имитации процесса работы необходимо выполнить команду меню щелкнуть кнопку включения питания на панели инструментов. Данное действие приведет в рабочее состояние схему и в одном из окон строки состояния будет показываться время работы схемы, которое не соответствует реальному и зависит от скорости процессора и системы персонального компьютера, именно поэтому для разработки сложных схем рекомендуется использовать компьютеры Pentium II с тактовой частотой процессора 266 МГц. Прервать имитацию можно двумя способами. Если вы закончили работу и просмотр результатов имитации можно повторно щелкнуть переключатель питания. Если же нужно временно прервать работу схемы, например, для детального рассмотрения осциллограммы, а затем продолжить работу, то можно воспользоваться кнопкой Pause, которая также расположена на панели инструментов. Возможность приостановки процесса также является большим достоинством по сравнению с традиционным тестированием радиоэлектронных устройств.

Теперь, для произведения анализа имитации можно изменять номиналы элементов, выводить и настраивать терминалы приборов. В данном случае можно просмотреть осциллограмму на выходе RC-цепи. Для этого нужно вывести окно терминала осциллографа двойным нажатием на компоненте Oscilloscope. Вид панели осциллографа представлен на рисунке 9.

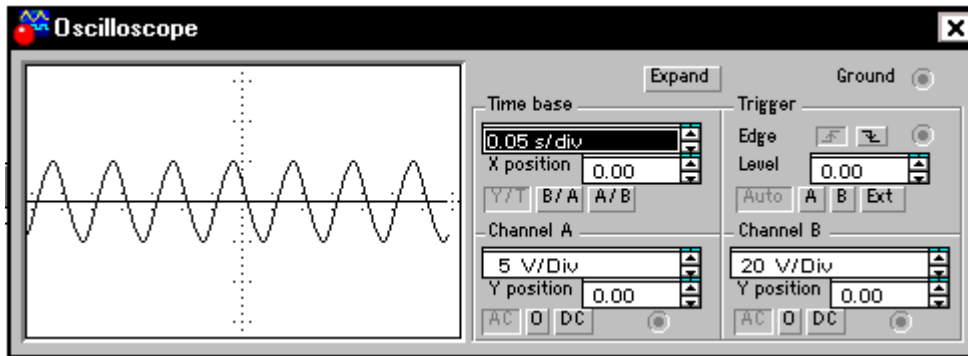


Рисунок 9 – Вид нормальной панели осциллографа Electronics Workbench

При использовании осциллографа в Electronics Workbench есть возможность просмотра сигнала на протяжении всего времени имитации. Для этого можно воспользоваться кнопкой Expand и воспользоваться полосами прокрутки изображения, чтобы перевести панель в нормальный режим используется кнопка Reduce.

Вид расширенной панели осциллографа показан на рисунке 10.

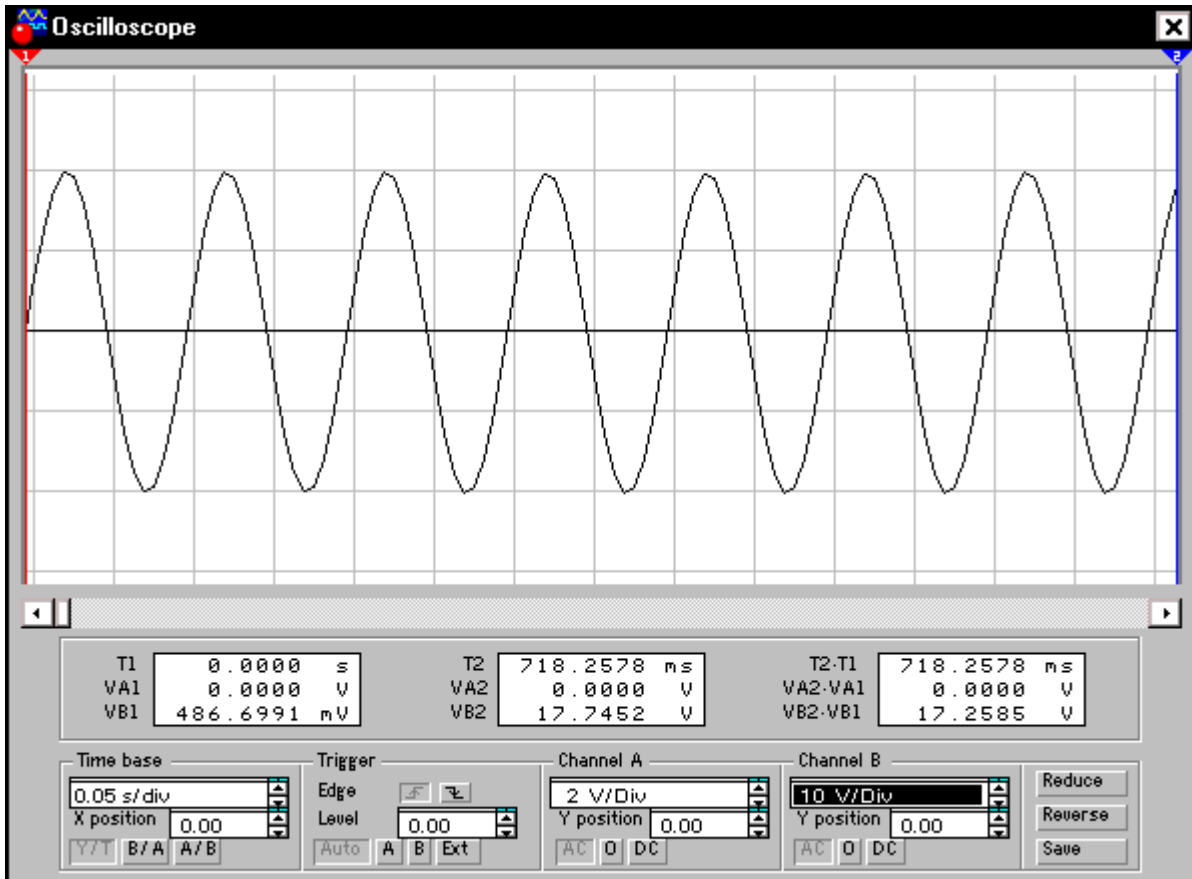


Рисунок 10 – Расширенная панель осциллографа

Теперь для изучения свойств RC-цепи можно изменить сигнал на ее входе. Для этого нужно вывести при помощи двойного нажатия кнопкой мыши

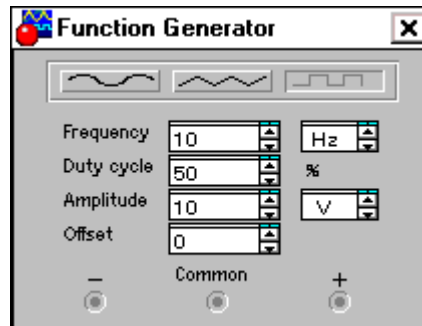


Рисунок 11 – Генератор импульсов

на компоненте на экран панель генератора импульсов. Ее вид представлен на рисунке 11.

При помощи генератора импульсов можно формировать три вида сигналов: синусоидальный, пилообразный и прямоугольный. В данном случае для анализа нужен прямоугольный импульс. Для перевода генератора в нужный режим требуется нажать соответствующую кнопку на панели. Также можно изменить другие параметры – частоту и амплитуду сигнала.

Перед изменением каких либо параметров следует отключать источники питания схемы, иначе возможно получение неверных результатов.

Выходной сигнал интегрирующей цепи показан на рисунке 12.

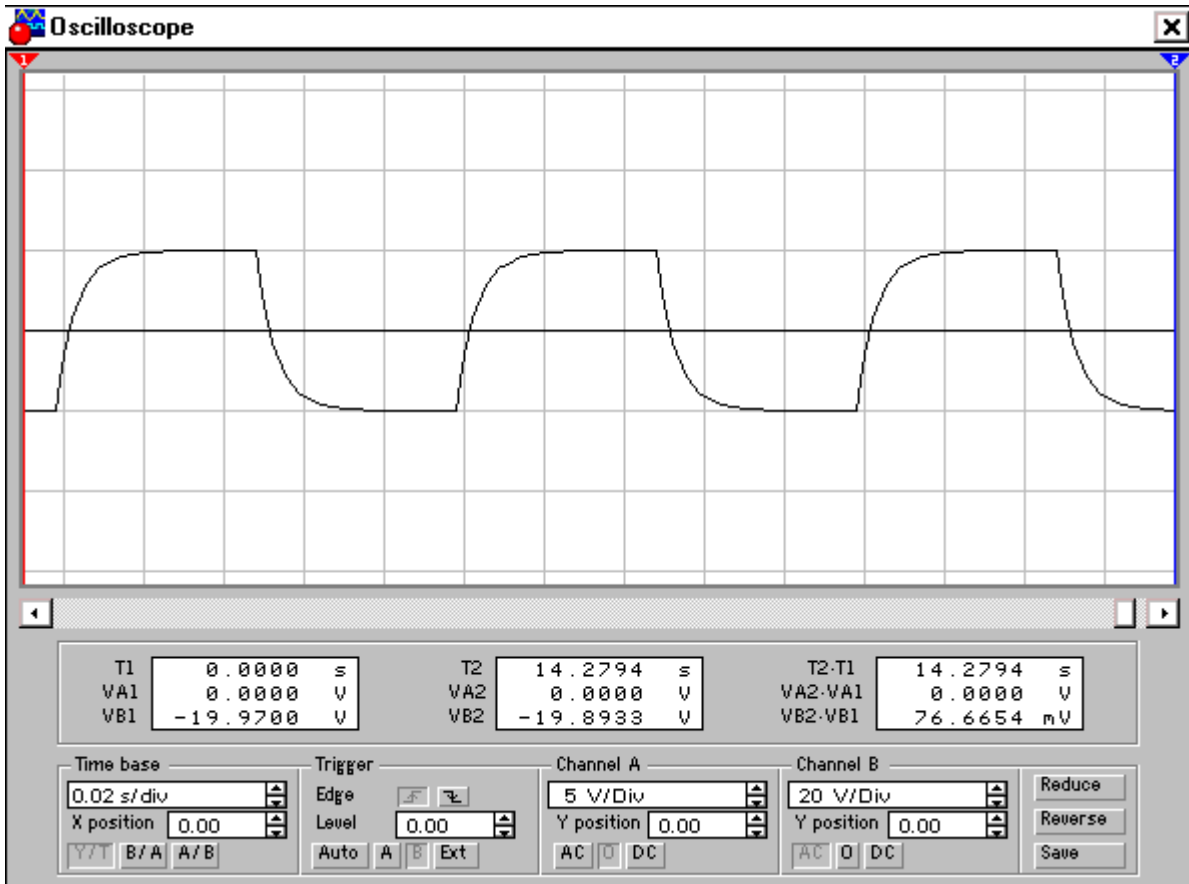


Рисунок 12– Сигнал на выходе интегрирующей RC – цепи

Для того, чтобы изменить какие-либо параметры элементов схемы нужно дважды щелкнуть левой кнопкой мыши на нужном элементе, при этом будет выведено окно свойств элемента. Пример такого окна приведен на рисунке 13.

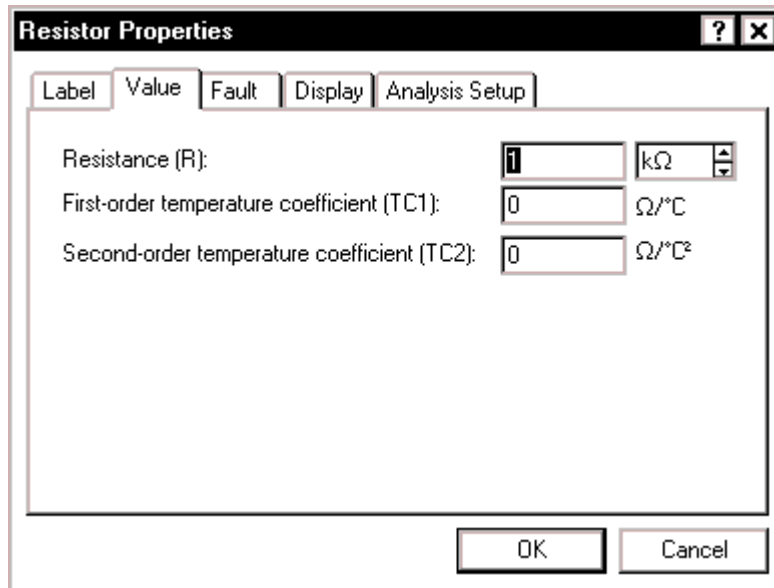


Рисунок 13 - Окно параметров резистора

Кроме анализа прямого наблюдения за терминалами инструментов Electronics Workbench позволяет выполнить дополнительные виды анализа. В качестве примера для данной схемы можно привести получение АЧХ и ФЧХ схемы как четырехполюсника. Т.е. при расчете на вход схемы будет подаваться сигнал различной частоты и будет произведен анализ зависимости вида выходного сигнала от входного. При этом нужно будет задать начальную и конечную частоты, на которых будет произведен анализ. Для проведения этого анализа нужно остановить работу цепи, т.е. воспользоваться переключателем питания или кнопкой Pause и выполнить команду меню Analysis / AC Frequency. Перед расчетом будет выведено окно параметров анализа. Вид этого окна приведен на рисунке 14. При необходимости можно изменить некоторые из параметров: Start frequency (начальная частота), End frequency (конечная частота), Sweep type (тип горизонтальной оси на конечном графике), Number of points (количество точек анализа). В данном случае удобно установить количество исследуемых точек равным 1000 для получения более гладкого графика, тип горизонтальной оси – логарифмическим и диапазон частот от 1Гц до 100 КГц.

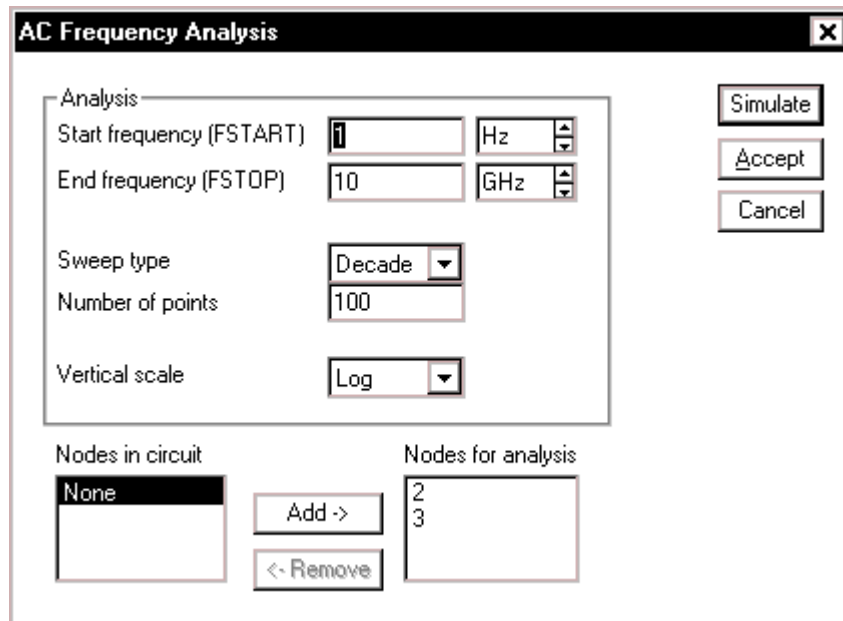


Рисунок 14 – Параметры анализа AC Frequency

Для получения графиков АЧХ и ФЧХ нужно нажать кнопку Simulate в окне параметров анализа, после чего будет выведено окно результатов представленное на рисунке 15.

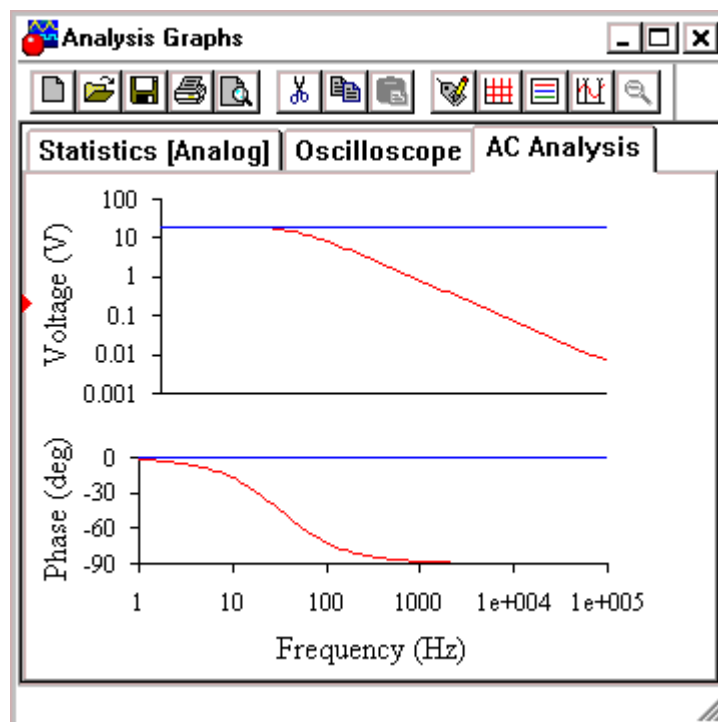


Рисунок 15 – АЧХ и ФЧХ интегрирующей RC – цепи

Для получения АЧХ и ФЧХ анализируемой цепи можно воспользоваться плоттером. На рисунке 16 показано подключение плоттера для измерения АЧХ. Для измерения ФЧХ следует нажать на плоттере кнопку Phase.

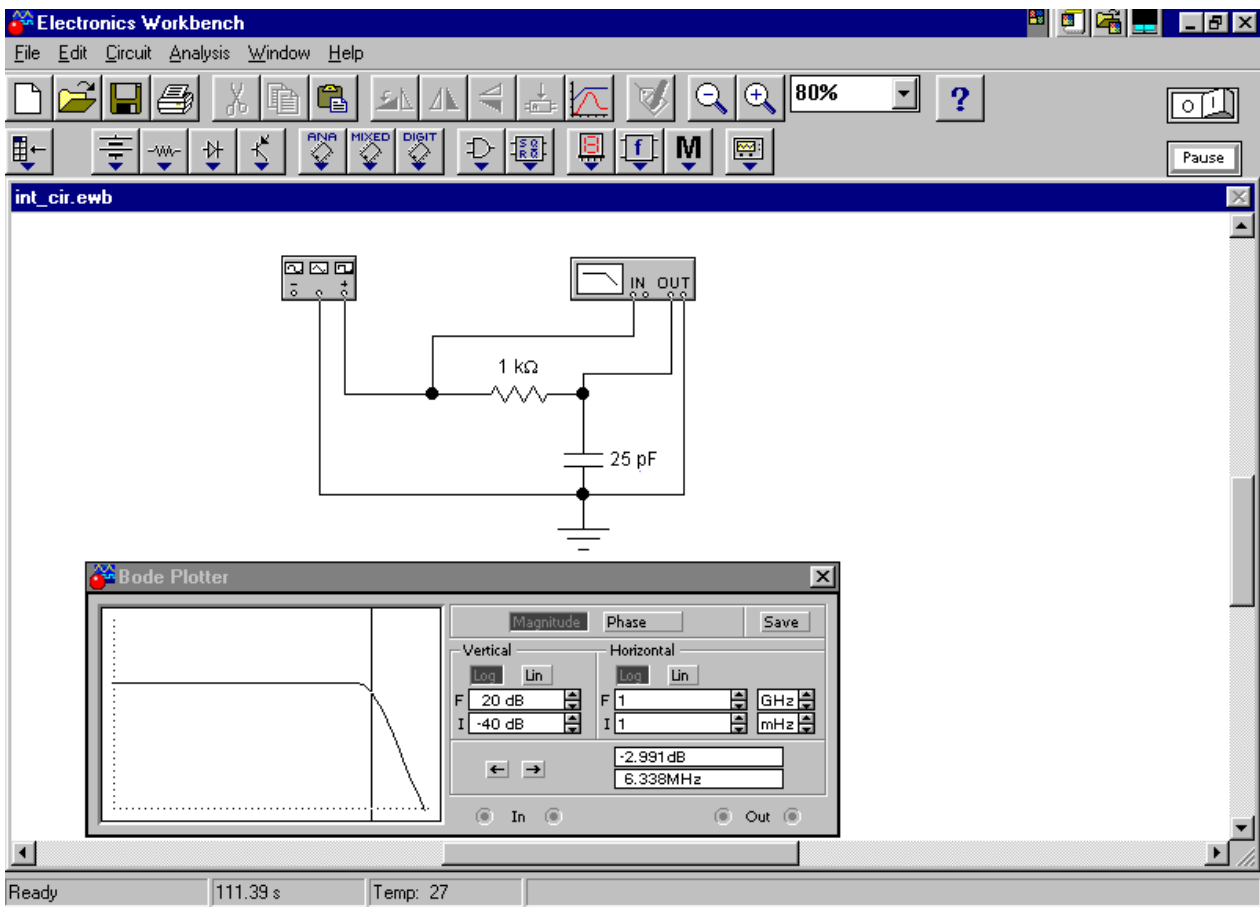


Рисунок 16 – Измерение АЧХ с помощью плоттера

Для решения возможных проблем, следует обращаться к Help программы, программный комплекс Electronics Workbench подробно описан в [2].

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 1×1 , 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

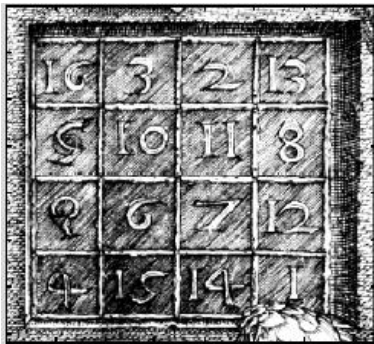
Вы можете вводить матрицы в MATLAB несколькими способами:

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 тоже 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 сослаться 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 сослаться 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 10^{308}

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 `sin`, 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.

π 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, $(2-e)2^{1023}$

`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

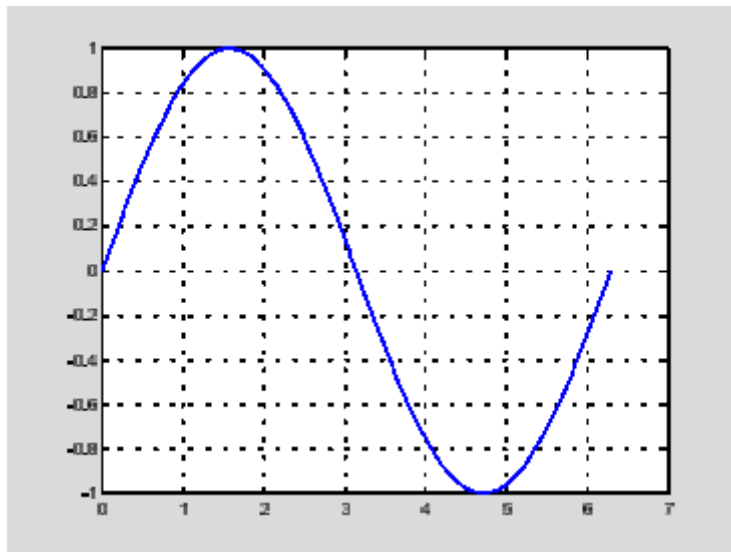
Функция `plot` имеет различные формы, связанные с входными параметрами, например `plot(y)` создает кусочно-линейный график зависимости элементов y от их индексов. Если вы зададите два вектора в качестве аргументов, `plot(x,y)` создаст график зависимости y от x .

Например, для построения графика значений функции \sin от нуля до 2π , сделаем следующее

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

`y = sin(t);`

`plot(t,y)`

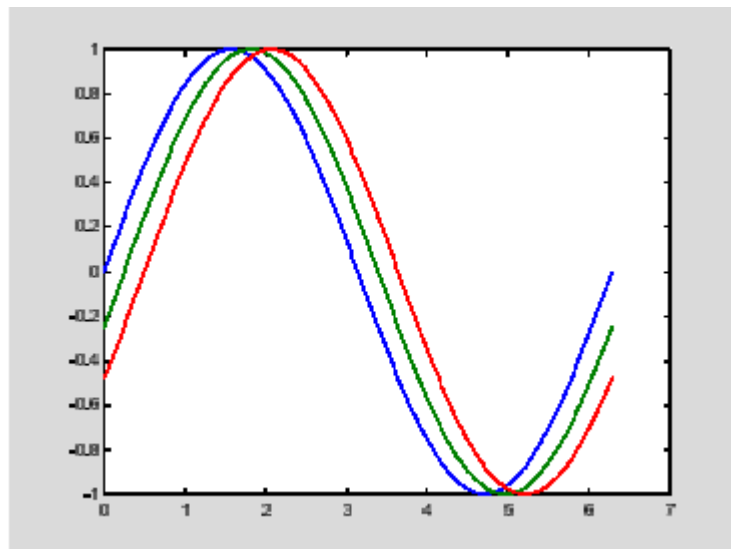


Вызов функции `plot` с многочисленными парами `x-y` создает многочисленные графики. MATLAB автоматически присваивает каждому графику свой цвет (исключая случаи, когда это делает пользователь), что позволяет различать заданные наборы данных. Например, следующие три строки отображают график близких функций,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

$$y2 = \sin(t-.25);$$

$$y3 = \sin(t-.5);$$

$$\text{plot}(t, y, t, y2, t, y3)$$



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

$$\text{plot}(x, y, \text{'цвет_стиль_маркер'})$$

`цвет_стиль_маркер` это 1-, 2-, 3-х символьная строка (заклученная в одинарные кавычки), составленная из типов цвета, стиля линий и маркеров:

Символы, относящиеся 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', '*' и '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 `rslog` 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 p -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;*

*[Ч,Х,Ж] = 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 x и y 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)
```

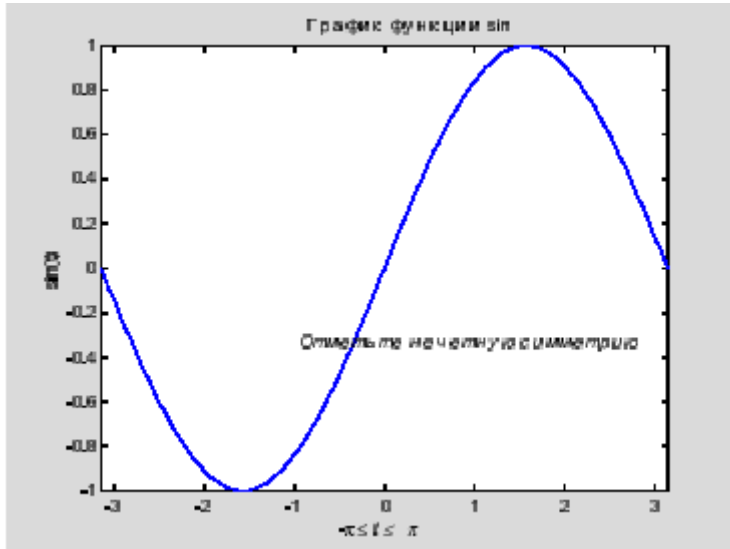
```
axis([-pi pi -1 1])
```

```
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 Y - вектор y.

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

```
[Y, X] = meshgrid(-8:.5:8);
```

```
R = sqrt(X.^2+Y.^2)+eps;
```

```
Z = sin(R)./R; mesh (Y, X, Z)
```

В этом примере R - это расстояние от начала координат, которому соответствует центр матрицы. Добавление 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 `map`). Элементы матрицы `X` - это целые числа от 1 до 128, которые служат индикаторами в цветном отображении, `map`. Следующие строки

```
imag(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 `magicsquare.eps`:

```
print -depsc2 magicsquare.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 10⁻³, 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 умещается 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 ... - 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 г. 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(N) is an N-by-N matrix constructed from the integers 1 through N² 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.

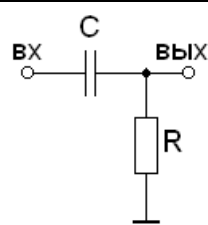
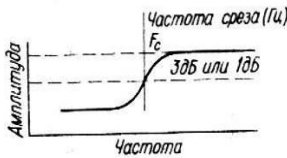
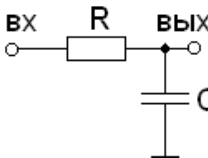
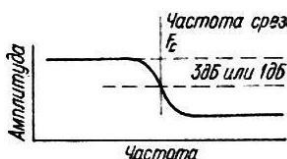
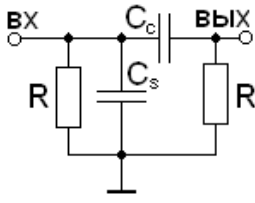
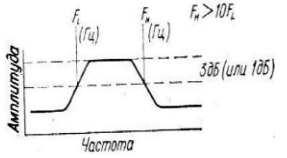
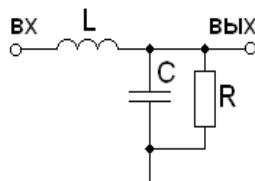
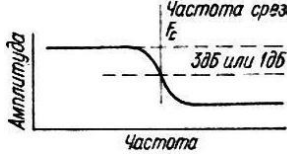
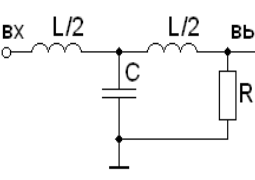
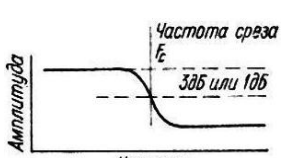
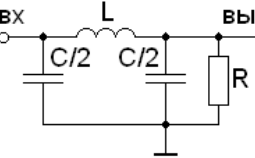
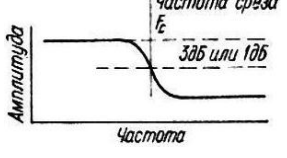
Материал подготовлен на основе:

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

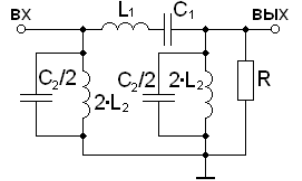
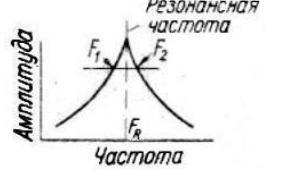
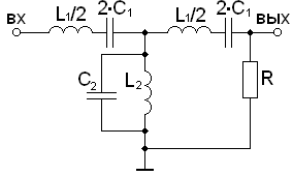

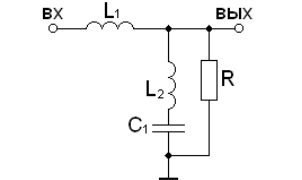
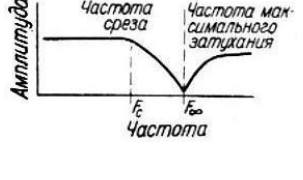
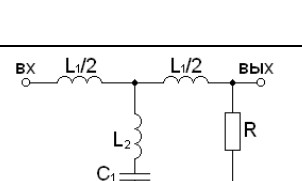

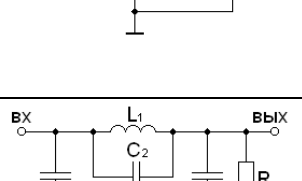

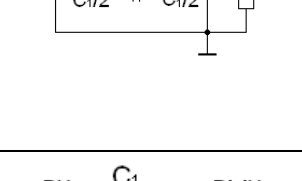
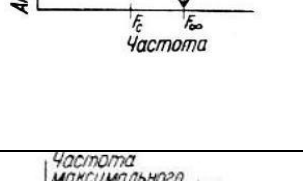
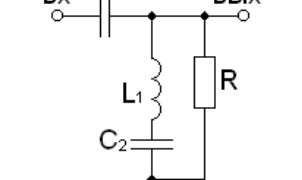
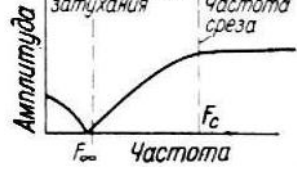
ПРИЛОЖЕНИЕ В

Индивидуальные задания

Таблица 1

№	Тип фильтра	Электрическая схема	Вид АЧХ	Расчетные соотношения
1	RC-фильтр ВЧ			$F_c(\tilde{A}\ddot{o}) \approx (6.28 \cdot RC)^{-1}$; $R(O\grave{i}) \approx (6.28 \cdot F_c C)^{-1}$; $C(\hat{O}) \approx (6.28 \cdot F_c R)^{-1}$; (для 3 дБ).
2	RC-фильтр НЧ			$F_c(\tilde{A}\ddot{o}) \approx (6.28 \cdot RC)^{-1}$; $R(O\grave{i}) \approx (6.28 \cdot F_c C)^{-1}$; $C(\hat{O}) \approx (6.28 \cdot F_c R)^{-1}$; (для 3 дБ).
3	Полосовой RC-фильтр			$F_L(\tilde{A}\ddot{o}) \approx (12.56 \cdot C_c R)^{-1}$; $F_H(\tilde{A}\ddot{o}) \approx (3.14 \cdot C_c R)^{-1}$; $C_s(\hat{O}) \approx (3.14 \cdot F_H R)^{-1}$; $C_c(\hat{O}) \approx (3.14 \cdot F_L R)^{-1}$; (для 3 дБ).
4	Г-обр. LC-фильтр НЧ типа К			$L \approx R / (3.14 \cdot F_c)^{-1}$; $C \approx (3.14 \cdot F_c R)^{-1}$; $F_c \approx (3.14 \cdot \sqrt{LC})^{-1}$; $R \approx \sqrt{L/C}$;
5	Т-обр. LC-фильтр НЧ типа К			$L \approx R / (3.14 \cdot F_c)^{-1}$; $C \approx (3.14 \cdot F_c R)^{-1}$; $F_c \approx (3.14 \cdot \sqrt{LC})^{-1}$; $R \approx \sqrt{L/C}$;
6	П-обр. LC-фильтр НЧ типа К			$L \approx R / (3.14 \cdot F_c)^{-1}$; $C \approx (3.14 \cdot F_c R)^{-1}$; $F_c \approx (3.14 \cdot \sqrt{LC})^{-1}$; $R \approx \sqrt{L/C}$;

7	Г-обр. LC-фильтр ВЧ типа К			$L \approx R / (12.6 \cdot F_C)^{-1};$ $C \approx (12.6 \cdot F_C R)^{-1};$ $F_C \approx (12.6 \cdot \sqrt{LC})^{-1};$ $R \approx \sqrt{L/C};$
8	Т-обр. LC-фильтр ВЧ типа К			$L \approx R / (12.6 \cdot F_C)^{-1};$ $C \approx (12.6 \cdot F_C R)^{-1};$ $F_C \approx (12.6 \cdot \sqrt{LC})^{-1};$ $R \approx \sqrt{L/C};$
9	Т-обр. LC-фильтр ВЧ типа К			$L \approx R / (12.6 \cdot F_C)^{-1};$ $C \approx (12.6 \cdot F_C R)^{-1};$ $F_C \approx (12.6 \cdot \sqrt{LC})^{-1};$ $R \approx \sqrt{L/C};$
10	Режкт. Г-обр. LC-фильтр			$L_1 \approx R(F_2 - F_1) / (3.14 \cdot F_1 F_2);$ $L_2 \approx R / [12.6 \cdot (F_2 - F_1)];$ $C_1 \approx [12.6 \cdot R(F_2 - F_1)]^{-1};$ $C_2 \approx (F_2 - F_1) / (3.14 \cdot F_1 F_2);$ $F_R(\epsilon \tilde{A} \tilde{\omega}) \approx 159 \cdot [L(i \epsilon \tilde{A} \tilde{t}) \cdot C(i \epsilon \tilde{O})]^{-\frac{1}{2}};$
11	Режкт. П-обр. LC-фильтр			$L_1 \approx R(F_2 - F_1) / (3.14 \cdot F_1 F_2);$ $L_2 \approx R / [12.6 \cdot (F_2 - F_1)];$ $C_1 \approx [12.6 \cdot R(F_2 - F_1)]^{-1};$ $C_2 \approx (F_2 - F_1) / (3.14 \cdot F_1 F_2);$ $F_R(\epsilon \tilde{A} \tilde{\omega}) \approx 159 \cdot [L(i \epsilon \tilde{A} \tilde{t}) \cdot C(i \epsilon \tilde{O})]^{-\frac{1}{2}};$
12	Режкт. Т-обр. LC-фильтр			$L_1 \approx R(F_2 - F_1) / (3.14 \cdot F_1 F_2);$ $L_2 \approx R / [12.6 \cdot (F_2 - F_1)];$ $C_1 \approx [12.6 \cdot R(F_2 - F_1)]^{-1};$ $C_2 \approx (F_2 - F_1) / (3.14 \cdot F_1 F_2);$ $F_R(\epsilon \tilde{A} \tilde{\omega}) \approx 159 \cdot [L(i \epsilon \tilde{A} \tilde{t}) \cdot C(i \epsilon \tilde{O})]^{-\frac{1}{2}};$
13	Полосовой Г-обр. LC-фильтр			$L_1 \approx R / [3.14 \cdot (F_2 - F_1)];$ $L_2 \approx R(F_2 - F_1) / (12.6 \cdot F_1 F_2);$ $C_1 \approx (F_2 - F_1) / [12.6 \cdot R F_1 F_2];$ $C_2 \approx [3.14 \cdot R(F_2 - F_1)]^{-1};$ $F_R(\epsilon \tilde{A} \tilde{\omega}) \approx 159 \cdot [L(i \epsilon \tilde{A} \tilde{t}) \cdot C(i \epsilon \tilde{O})]^{-\frac{1}{2}};$

14	Полосовой П-обр. LC-фильтр			$L_1 \approx R/[3.14 \cdot (F_2 - F_1)];$ $L_2 \approx R(F_2 - F_1)/(12.6 \cdot F_1 F_2);$ $C_1 \approx (F_2 - F_1)/[12.6 \cdot R F_1 F_2];$ $C_2 \approx [3.14 \cdot R(F_2 - F_1)]^{-1};$ $F_R(\epsilon \tilde{A} \tilde{\omega}) \approx 159 \cdot [L(i \epsilon \tilde{A} i) \cdot C(i \epsilon \tilde{\omega})]^{-\frac{1}{2}};$
15	Полосовой Т-обр. LC-фильтр			$L_1 \approx R/[3.14 \cdot (F_2 - F_1)];$ $L_2 \approx R(F_2 - F_1)/(12.6 \cdot F_1 F_2);$ $C_1 \approx (F_2 - F_1)/[12.6 \cdot R F_1 F_2];$ $C_2 \approx [3.14 \cdot R(F_2 - F_1)]^{-1};$ $F_R(\epsilon \tilde{A} \tilde{\omega}) \approx 159 \cdot [L(i \epsilon \tilde{A} i) \cdot C(i \epsilon \tilde{\omega})]^{-\frac{1}{2}};$
16	Г-обр. LC-фильтр НЧ м-типа			$m = \sqrt{1 - (F_C / F_\infty)^2} = const;$ $C_1 = m/(3.14 \cdot F_C R);$ $C_2 = (1 - m^2)/(12.6 \cdot m F_C R);$ $L_1 = mR/(3.14 \cdot F_C);$ $L_2 = (1 - m^2)R/(12.6 \cdot m F_C);$
17	Т-обр. LC-фильтр НЧ м-типа			$m = \sqrt{1 - (F_C / F_\infty)^2} = const;$ $C_1 = m/(3.14 \cdot F_C R);$ $C_2 = (1 - m^2)/(12.6 \cdot m F_C R);$ $L_1 = mR/(3.14 \cdot F_C);$ $L_2 = (1 - m^2)R/(12.6 \cdot m F_C);$
18	П-обр. LC-фильтр НЧ м-типа			$m = \sqrt{1 - (F_C / F_\infty)^2} = const;$ $C_1 = m/(3.14 \cdot F_C R);$ $C_2 = (1 - m^2)/(12.6 \cdot m F_C R);$ $L_1 = mR/(3.14 \cdot F_C);$ $L_2 = (1 - m^2)R/(12.6 \cdot m F_C);$
19	Г-обр. LC-фильтр ВЧ м-типа			$m = \sqrt{1 - (F_\infty / F_C)^2} = const;$ $C_1 = (12.6 \cdot m F_C R)^{-1};$ $C_2 = m/[3.14 \cdot R F_C (1 - m^2)];$ $L_1 = R/(12.6 \cdot m F_C);$ $L_2 = mR/[3.14 \cdot F_C (1 - m^2)];$
20	Т-обр. LC-фильтр ВЧ м-типа			$m = \sqrt{1 - (F_\infty / F_C)^2} = const;$ $C_1 = (12.6 \cdot m F_C R)^{-1};$ $C_2 = m/[3.14 \cdot R F_C (1 - m^2)];$ $L_1 = R/(12.6 \cdot m F_C);$ $L_2 = mR/[3.14 \cdot F_C (1 - m^2)];$

21	П-обр. LC-фильтр ВЧ m-типа			$m = \sqrt{1 - (F_\infty / F_C)^2} = const;$ $C_1 = (12.6 \cdot m F_C R)^{-1};$ $C_2 = m / [3.14 \cdot R F_C (1 - m^2)];$ $L_1 = R / (12.6 \cdot m F_C);$ $L_2 = m R / [3.14 \cdot F_C (1 - m^2)];$
22	Т-обр. LC-фильтр НЧ m-типа с посл. резонанс.			$F_C = [3.14 \cdot C_2 (L_1 + 4 \cdot L_2)]^{-1};$ $F_\infty = (6.28 \cdot \sqrt{L_2 C_2})^{-1};$
23	П-обр. LC-фильтр НЧ m-типа с посл. резонанс.			$F_C = [3.14 \cdot C_2 (L_1 + 4 \cdot L_2)]^{-1};$ $F_\infty = (6.28 \cdot \sqrt{L_2 C_2})^{-1};$
24	П-обр. LC-фильтр НЧ m-типа с паралл. резонанс.			$F_C = [3.14 \cdot \sqrt{L_1 (C_2 + 4 \cdot C_1)}]^{-1};$ $F_\infty = (6.28 \cdot \sqrt{L_1 C_1})^{-1};$
25	Т-обр. LC-фильтр НЧ m-типа с паралл. резонанс.			$F_C = [3.14 \cdot \sqrt{L_1 (C_2 + 4 \cdot C_1)}]^{-1};$ $F_\infty = (6.28 \cdot \sqrt{L_1 C_1})^{-1};$
26	Т-обр. LC-фильтр ВЧ m-типа с посл. резонанс.			$F_C = [12.6 \cdot \sqrt{(L_2 C_1)^{-1} + 4 \cdot (L_2 C_2)^{-1}}]^{-1};$ $F_\infty = (6.28 \cdot \sqrt{L_2 C_2})^{-1};$
27	П-обр. LC-фильтр ВЧ m-типа с посл. резонанс			$F_C = [12.6 \cdot \sqrt{(L_2 C_1)^{-1} + 4 \cdot (L_2 C_2)^{-1}}]^{-1};$ $F_\infty = (6.28 \cdot \sqrt{L_2 C_2})^{-1};$
28	Т-обр. LC-фильтр ВЧ m-типа с паралл. резонанс.			$F_C = [12.6 \cdot \sqrt{(L_2 C_1)^{-1} + 4 \cdot (L_1 C_1)^{-1}}]^{-1};$ $F_\infty = (6.28 \cdot \sqrt{L_1 C_1})^{-1};$
29	П-обр. LC-фильтр ВЧ m-типа с паралл. резонанс.			$F_C = [12.6 \cdot \sqrt{(L_2 C_1)^{-1} + 4 \cdot (L_1 C_1)^{-1}}]^{-1};$ $F_\infty = (6.28 \cdot \sqrt{L_1 C_1})^{-1};$

ПРИЛОЖЕНИЕ Г

Бланк задания и требования к отчету

УТВЕРЖДАЮ
Зав. каф. РЗИ _____ А.С.Задорин
_____ 200_г.

ВАРИАНТ ЗАДАНИЯ №__

На летнюю вычислительную практику

студенту гр. _____

Тема практики: работа с пакетами MatLAB и Electronics Work Bench (EWB).

Содержание практики: *формирование и решение системы линейных алгебраических уравнений, моделирующих электрическую цепь, создание программы расчета АЧХ и ФЧХ фильтрующих RLC электрических цепей, проверка результатов с помощью пакета EWB.*

Содержание работы: знакомство с системой MatLAB, реализация алгоритма, предложенного преподавателем, проверка результата, оформление пояснительной записки и защита работы. При реализации предложенного алгоритма следует максимально использовать возможности системы MatLAB (кнопочное меню, графику, чтение данных из файла, формирование файла результатов и т.д.). Пояснительную записку выполнить с помощью редактора Word с **обязательным** соблюдением всех требований стандарта предприятия СТП ТУСУР.

Обязательные пункты пояснительной записки:

- ♣ Титульный лист.
- ♣ Реферат.
- ♣ Задание.
- ♣ Содержание.
- ♣ Введение.
- ♣ Основные содержательные разделы и подразделы.
- ♣ Заключение.
- ♣ Список используемой литературы.
- ♣ Приложения.

Рекомендуемая литература:

- ◆ Любая по системе MatLAB, например, Лазарев Ю.Ф. **MatLAB 5.x** – К.:ВНУ,2000.-384с. (Серия «Библиотека студента»).
- ◆ Методические материалы, указанные преподавателем.

Преподаватель _____
_____ 200_г.

Отчет подлежит сдаче до _____ 200_г.

Студент _____

Требования к отчету по «Вычислительной практике».

1. Основной текст с приложениями (не менее 10 страниц через 1 интервал) набирается в редакторе Word-2003 и представляется в одном экземпляре на CD;
2. Формат страницы - А4, правое, левое и нижнее поля - по 22 мм, а верхнее - 30 мм; колонтитул верхний – 1.5, нижний – 0;
3. Основной материал набирается одним и тем же шрифтом Times New Roman размером 14. Заголовки разделов оформляются жирными заглавными буквами шрифтом Arial размером 14 и выравниваются по центру. Заголовки подразделов набираются жирными прописными буквами шрифтом Arial размером 14 и имеют отступ красной строки.
4. Отступ красной строки — 1,27 см
5. Основной текст должен быть выровнен по ширине, опция переноса слов выключена.
6. Страницы нумеруются начиная со второй страницы содержания, вверху по центру, размер шрифта- 14.
7. Формулы (если есть) выравниваются по центру и нумеруются в круглых скобках (номера формул выравниваются по правому краю страницы).
8. Ссылки на литературу выполняются в квадратных скобках (например, [1]). В списке литературы ссылки располагаются по мере указания в тексте.
9. Рисунки выполняются средствами Word, либо любым доступным графическим редактором с последующей вставкой в текст. Нумерация рисунков должна быть самостоятельной внутри каждого раздела (например, Рисунок 1.1 – первый рисунок первого раздела). Все рисунки должны содержать подрисуночные подписи.
10. Таблицы следует вставлять непосредственно в текст. Таблица сверху и снизу отделяется от основного текста одной пустой строкой. Нумерация таблиц самостоятельная внутри каждого раздела (например, Таблица 1.1 первая таблица первого раздела). Если в состав материала входит только одна таблица она не нумеруется.
11. Ссылки на таблицы и рисунки даются с сокращениями слов "таблица" и "рисунок" (например: в табл.1.1, приведены результаты, на рис.2.1 показана зависимость

Обязательные пункты пояснительной записки.

- ♣ Титульный лист.
- ♣ Реферат.
- ♣ Задание.
- ♣ Содержание (как правило, с этой страницы, проставляются номера страниц).

- ♣ Введение (раздел не нумеруется).
- ♣ Основные содержательные разделы и подразделы (разделы нумеруются 1, 2, и т.д.; подразделы – 1.1, 1.2 и т.д.).
- ♣ Заключение (раздел не нумеруется).
- ♣ Список используемой литературы.
- ♣ Приложения (нумеруются заглавными буквами русского алфавита: Приложение А, Приложение Б и т.д.). В приложении могут присутствовать листинги программ с комментариями; структурные схемы программ, результат тестирования программ, пример результатов вычислений (если есть вычисления), необходимый справочный материал и т.д.

Оформление некоторых листов пояснительной записки можно посмотреть в кабинете курсового и дипломного проектирования кафедры РЗИ(ауд.416 РК).

ПРИЛОЖЕНИЕ Д

Примеры m – файлов

```
function [n,vv]=ytop(dd);
%определение числа узлов n и числа ветвей vv
%dd-входной массив-вектор данных об элементах схемы
nm=max(size(dd));%определение размерности массива dd
k=1;n=0;vv=0;
while k<=nm;%перебор элементов схемы с шагом kp
    pr=upper(dd{k}); %выделение признака элемента схемы (признак может
%быть задан в любом регистре)
    if (pr=='G')|(pr=='L')|(pr=='C')|(pr=='I');%проводимость,
%индуктивность, емкость, источник тока
        kd=k+3;kp=4;vv=vv+1;
    end;
    bn=k+2;
    for k4=bn:kd;%определение n путем выбора макс. номера узла
        if ceil(dd{k4})>n;
            n=ceil(dd{k4});
        end;
    end;
    k=k+kp;
end;
```

```
function [t,ww]=ymatr(dd,n,w);
%формирование Y-матрицы
mn=max(size(dd));
t=zeros(n,n);
ww=zeros(n,1);
k=1;du=1;
while k<=mn;
    pr=upper(dd{k});
    kp=4;
    %Емкость-C
    if pr=='C';
        n1=ceil(dd{k+2});n2=ceil(dd{k+3});zn=abs(dd{k+1});
        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';
        n1=ceil(dd{k+2});n2=ceil(dd{k+3});zn=abs(dd{k+1});
```

```

    if n1~=0;
        t(n1,n1)=t(n1,n1)+1/zn;
    end;
    if n2~=0;
        t(n2,n2)=t(n2,n2)+1/zn;
    end;
    if (n1~=0) & (n2~=0);
        t(n1,n2)=t(n1,n2)-1/zn;t(n2,n1)=t(n2,n1)-1/zn;
    end;
end;
%Индуктивность-L
if pr=='L';
    n1=ceil(dd{k+2});n2=ceil(dd{k+3});zn=abs(dd{k+1});
    if n1~=0;
        t(n1,n1)=t(n1,n1)+1/(j*zn*w);
    end;
    if n2~=0;
        t(n2,n2)=t(n2,n2)+1/(j*zn*w);
    end;
    if (n1~=0) & (n2~=0);
        t(n1,n2)=t(n1,n2)-1/(j*zn*w);t(n2,n1)=t(n2,n1)-1/(j*zn*w);
    end;
end;
%Источник тока-I
if pr=='I';
    n1=ceil(dd{k+2});n2=ceil(dd{k+3});zn=abs(dd{k+1});
    if n1~=0;
        ww(n1,1)=ww(n1,1)+zn;
    end;
    if n2~=0;
        ww(n2,1)=ww(n2,1)-zn;
    end;
end;
%Переход к следующему элементу
k=k+kp;
end;

clc;
disp('    Mein    ')
disp('    МУПр частотный анализ    ')
disp('Проводимость-G : в схеме номинал задавать в омах!!!')
disp('Емкость-C')
disp('Индуктивность-L')
disp('Источник тока-I')
%
%Инт. цепь с C : fn=1e5 fk=1e7 ni=1 no=2
dd={'I' 1 1 0 'G' 1e3 1 2 'C' 25e-12 2 0};
%
ni=input('Номер входного узла= ');
no=input('Номер выходного узла= ');
fn=input('Нач. частота= ');

```

```

fk=input('Кон. частота= ');
kf=input('Количество точек = ');
df=(fk-fn)/kf;
[n,vv]=ytop(dd)
f=fn;k1=1;
while f<=fk;
    w=2*pi*f;
    [t,ww]=ymatr(dd,n,w);
    uv=t\ww;
    uv1=uv(ni);uv2=uv(no);
    ku=uv2/uv1;mf(k1)=f;mku(k1)=abs(ku);
    ph(k1)=angle(ku)*180/pi;
    k1=k1+1;
    f=f+df;
end;
%Вывод ЧХ
k2=1:k1-1;
subplot(2,1,1);
semilogx(mf(k2),mku(k2));
title('АЧХ');
xlabel('
f,Гц');
ylabel('Ku,отн.ед. ');
grid
subplot(2,1,2);
semilogx(mf(k2),ph(k2));
title('ФЧХ');
xlabel('
f,Гц');
ylabel('Phase,grad');
grid
disp('      End Mein      ')

```