

Лабораторная работа №1 по курсу «Радиотехнические устройства и системы» Введение в GNU/Octave

Кузнецов В.В., ассистент кафедры ЭИУ1-КФ

6 сентября 2013 г.

1 Цель работы

Целью семинара является генерация звуковых сигналов с использованием звуковой карты персонального компьютера и системы численной математики GNU/Octave.

Система GNU/Octave — это высокоуровневый язык программирования, предназначенный прежде всего для численных расчётов. Он предоставляет удобный интерфейс командной строки для численного решения линейных и нелинейных задач, а также для выполнения других численных экспериментов. С помощью GNU/Octave можно решать задачи в том числе генерации и обработки сигналов. Установить GNU/Octave для Linux можно в один клик через пакетный менеджер, а для Windows её можно бесплатно скачать с сайта разработчика <http://octave.sourceforge.net>.

Octave работает в режиме командной строки. Octave позволяет выполнять операции с действительными и комплексными числами, матрицами, решать системы линейных уравнений. Синтаксис команд Octave близок к языку C и повторяет среду Matlab.

В ходе выполнения лабораторной работы необходимо проделать в среде Octave задание, приведённые ниже.

2 Применение Octave для расчётов

2.1 Запуск и выход из программы

В состав пакета входит интерактивный командный интерфейс (интерпретатор). Интерпретатор Octave запускается из терминала ОС Linux или из его порта в Windows. После запуска Octave пользователь видит окно интерпретатора (см. рис. 1). Чтобы запустить GNU/Octave нужно для Linux набрать в терминале `octave`, а для Windows щёлкнуть по ярлыку, создаваемому на рабочем столе после установки программы.

Чтобы выйти из программы в командной строке нужно набрать `quit`. После того как команда напечатана, нужно нажать Enter.

2.2 Комментарии и вывод результата вычислений

В окне интерпретатора пользователь может вводить как отдельные команды языка Octave, так и группы команд, объединяемые в программы. Если строка заканчивается

```
GNU Octave, version 3.6.4
Copyright (C) 2013 John W. Eaton and others.
This is free software; see the source code for copying conditions.
There is ABSOLUTELY NO WARRANTY; not even for MERCHANTABILITY or
FITNESS FOR A PARTICULAR PURPOSE. For details, type 'warranty'.
```

```
Octave was configured for "x86_64-suse-linux-gnu".
```

```
Additional information about Octave is available at http://www.octave.org.
```

```
Please contribute if you find this software useful.
For more information, visit http://www.octave.org/get-involved.html
```

```
Read http://www.octave.org/bugs.html to learn how to submit bug reports.
```

```
For information about changes from previous versions, type 'news'.
```

```
octave:1>
```

Рис. 1. Командная строка Octave

символом ; (точка с запятой) результаты на экран не выводятся. Если же в конце строки символ отсутствует, результаты работы выводятся на экран (см. рис. 1.2). Текст в строке после символа % (процент) или # (решётка) является комментарием и интерпретатором не обрабатывается. Комментарии можно вводить и в командном режиме. Рассмотрим несколько несложных примеров.

2.3 Арифметические операции

Рассчитаем сколько будет, если сложить 2 и 3. Для этого введём в командной строке Octave следующие команды:

```
octave:1> 2+3
ans = 5
```

И получим в результате 5. В переменной `ans` хранится результат последней операции.

В GNU/Octave можно выполнять и операции с логарифмами, комплексными числами, тригонометрическими функциями. Например:

Логарифмирование:

$$\frac{\log_{10} 100}{\log_{10} 10} \quad (1)$$

```
octave:1> log10(100)/log10(10)
ans = 2
```

Тригонометрические функции (тангенс):

$$\left\lfloor \frac{1 + \tan 1.2}{1.2} \right\rfloor \quad (2)$$

```
octave:3> floor((1+tan(1.2)) / 1.2)
ans = 2
```

Корень квадратный и возведение в степень:

$$\sqrt{3^2 + 4^2} \quad (3)$$

```
octave:4> sqrt(3^2 + 4^2)
ans = 5
```

Экспоненты, комплексные числа и число π .

$$e^{i\pi} \quad (4)$$

```
octave:5> exp(i*pi)
ans = -1.0000e+00 + 1.2246e-16i
```

2.4 Пакетный и терминальный режимы работы Octave

Возможны два варианта решения любой задачи в Octave:

1. Терминальный режим. В этом режиме в окно интерпретатора последовательно вводятся отдельные команды.

2. Программный режим. В этом режиме создаётся текстовый файл с расширением `.m`, в котором хранятся последовательно выполняемые команды Octave. Затем этот текстовый файл (программа на языке Octave) запускается на выполнение в среде Octave.

Рассмотрим пример решения задач в обоих режимах.

Рассчитаем резонансную частоту LC - контура по формуле:

$$f = \frac{1}{2\pi\sqrt{LC}} \quad (5)$$

Где L — индуктивность катушки, Гн;

— ёмкость конденсатора, Ф;

f — частота, Гц

Чтобы решить нашу задачу в терминальном режиме введём последовательно после того как Octave запустится и выдаст приглашение, подобное рис.1, команды, показанные на рис. 2. На рисунке приведён вывод для Linux, отличие для системы Windows заключается только в способе запуска программы.

В переменной `ans` хранится результат последней операции, если команда не содержит знака присваивания. Следует помнить, что значение переменной `ans` изменяется после каждого вызова команды без операции присваивания.

Теперь рассмотрим, как решить эту же задачу в программном режиме. Вызовем любой текстовый редактор, например `kwrite` в системе Linux, но подойдёт даже обычный Notepad для Windows, в окне которого последовательно введём следующие команды, приведённые в листинге 1.

```

vvk@linux-bmx0:~> octave
GNU Octave, version 3.6.4
Copyright (C) 2013 John W. Eaton and others.
This is free software; see the source code for copying conditions.
There is ABSOLUTELY NO WARRANTY; not even for MERCHANTABILITY or
FITNESS FOR A PARTICULAR PURPOSE. For details, type 'warranty'.

Octave was configured for "x86_64-suse-linux-gnu".

Additional information about Octave is available at http://www.octave.org.

Please contribute if you find this software useful.
For more information, visit http://www.octave.org/get-involved.html

Read http://www.octave.org/bugs.html to learn how to submit bug reports.

For information about changes from previous versions, type 'news'.

octave:1> L=10e-6;
octave:2> C=47e-12;
octave:3> f=1/(2*pi*sqrt(L*C))
f = 7.3413e+06
octave:4> quit
vvk@linux-bmx0:~>

```

Рис. 2. Расчёт резонансной частоты LC-контурa в командной строке Octave

Листинг 1. Файл LC-kontur.m

```

1 #!/usr/bin/octave -qf # Первая строка --- тип интерпретатора
2 L=10e-6; # Индуктивность катушки, Гн
3 C=47e-12; # Ёмкость конденсатора, Ф
4 f=1/(2*pi*sqrt(L*C)) # Считаем резонансную частоту и
5 # выводим её на экран
6 # pause # Раскомментируйте строку для Windows. Задерживает выполнение
   скрипта
7 # до тех пор пока пользователь не нажмёт Enter.

```

Сохраним введённые команды в виде файла LC-kontur.m. Скрипты Octave обычно имеют расширение *.m. Теперь эту программу необходимо запустить на выполнение. В системе Linux нужно сначала сделать файл исполняемым любым способом, например из командной строки или файлового менеджера. Чтобы дать скрипту права на исполнение и затем запустить его нужно выполнить в каталоге, где находится файл скрипта следующие команды:

```

chmod a+x LC-kontur.m
./LC-kontur

```

В результате выполнения команд в окне терминала напечатается вычисленное значение резонансной частоты $f = 7.3413e+06$.

При обработке сигналов нам понадобится выполнять операции с одномерными массивами чисел с плавающей точкой (float). Одномерные массивы в Octave являются частным случаем матриц. Им соответствуют матрица-строка и матрица-столбец (вектор). Двухмерные матрицы — это двухмерные массивы. Индексация массивов начинается с единицы, в отличие от языка C.

Чтобы объявить массив нужно перечислить все его элементы через запятую и заключить их в квадратные скобки. Ниже объявлен массив X из трёх элементов и выполнено обращение к его второму элементу:

```
octave:1> X=[1,2,3]
X =
```

```
    1    2    3
```

```
octave:2> X(2)
ans = 2
```

Чтобы сгенерировать массив, содержащий последовательность чисел с некоторым шагом нужно воспользоваться операцией `:`. Например:

```
octave:3> X=0:1:10
X =
```

```
    0    1    2    3    4    5    6    7    8    9   10
```

```
octave:4> X(6)
ans = 5
```

Мы создали массив из целых чисел от 0 до 10 с шагом 1 и обратились к его шестому элементу.

Для построения графиков в Octave служит функция `plot(x,y)`. В качестве аргументов ей передаются два вектора (матрица-столбец, либо матрица-строка), содержащие точки по осям X и Y . Количество элементов в векторах должно быть одинаковым.

К матрицам и массивам в Octave можно применять все операции, известные из курса линейной алгебры: транспонирование, умножение на число, сложение, вычитание, умножение матриц, вычисление обратной матрицы.

Здесь нужно упомянуть про операцию транспонирования `'` (одинарная кавычка). Поместив знак `'` после матрицы мы заменяем строки на столбцы. Так можно получить из матрицы-строки матрицу-столбец.

Матрицы в Octave можно складывать так же как и обычные числа при помощи операции `+`. При этом первый элемент складывается с первым, второй со вторым и т.д. Этим пользуются для сложения сигналов.

Для примера построим график функции $\sin(t)$ на отрезке от 0 до 2π по 20 точкам. Команды, вводимые в окно Octave показаны на рис.3.

Если ввести в Octave указанные команды, то откроется графическое окно, в котором будет построена синусоида.

Как видно из простейших примеров, у Octave достаточно широкие возможности, а по синтаксису он близок к Matlab.

```

GNU Octave, version 3.6.4
octave:1> t=0:(2*pi/20):2*pi # генерируем вектор из 20 точек от 0 до 2*pi
t =

Columns 1 through 9:

    0.00000    0.31416    0.62832    0.94248    1.25664    1.57080    1.88496
2.19911    2.51327

Columns 10 through 18:

    2.82743    3.14159    3.45575    3.76991    4.08407    4.39823    4.71239
5.02655    5.34071

Columns 19 through 21:

    5.65487    5.96903    6.28319

octave:2> s=sin(t) # вычисляем таблицу функции sin()
s =

Columns 1 through 10:

0.00000    0.30902    0.58779    0.80902    0.95106    1.00000    0.95106    0.80902
0.58779    0.30902

Columns 11 through 20:

0.00000   -0.30902   -0.58779   -0.80902   -0.95106   -1.00000   -0.95106   -0.80902
0.58779   -0.30902

Column 21:

-0.00000

octave:3> plot(t,s); # строим график
octave:4>

```

Рис. 3. Пример построения графика функции в GNU/Octave