

Министерство образования Республики Беларусь  
Учреждение образования «Витебский государственный  
университет имени П.М. Машерова»  
Кафедра инженерной физики

**Е.А. Краснобаев, Н.С. Буйнов**

# **СИСТЕМЫ СВЯЗИ И СЕТИ ПЕРЕДАЧИ ИНФОРМАЦИИ**

*Методические рекомендации  
к выполнению лабораторных работ*

*Витебск  
ВГУ имени П.М. Машерова  
2017*

УДК 004.451.52(076.5)  
ББК 32.811.2я73  
К78

Печатается по решению научно-методического совета учреждения образования «Витебский государственный университет имени П.М. Машерова». Протокол № 2 от 28.12.2016 г.

Авторы: заведующий кафедрой инженерной физики ВГУ имени П.М. Машерова, кандидат технических наук, доцент **Е.А. Краснобаев**; доцент кафедры инженерной физики ВГУ имени П.М. Машерова, кандидат физико-математических наук **Н.С. Буйнов**

Рецензент:  
заведующий кафедрой прикладного и системного программирования ВГУ имени П.М. Машерова,  
кандидат физико-математических наук *С.А. Ермоченко*

**Краснобаев, Е.А.**

**К78** Системы связи и сети передачи информации : методические рекомендации к выполнению лабораторных работ / Е.А. Краснобаев, Н.С. Буйнов. – Витебск : ВГУ имени П.М. Машерова, 2017. – 32 с.

Методические рекомендации к выполнению лабораторных работ по дисциплине «Системы связи и сети передачи информации» предназначены для студентов специальностей 1-98 01 01 Компьютерная безопасность (по направлениям) и 1-31 04 01-02 Физика (производственная деятельность) и составлены в соответствии с учебной программой указанного курса.

УДК 004.451.52(076.5)  
ББК 32.811.2я73

© Краснобаев Е.А., Буйнов Н.С., 2017  
© ВГУ имени П.М. Машерова, 2017

## СОДЕРЖАНИЕ

Введение .....	4
Лабораторная работа № 1. Работа в среде Matlab. Структура Signal Processing Toolbox. Генерация сигналов .....	5
Лабораторная работа № 2. Исследование свойств преобразования Фурье дискретных сигналов .....	12
Лабораторная работа № 3. Цифровая модуляция сигналов .....	15
Лабораторная работа № 4. Канальное кодирование сигналов .....	24
Лабораторная работа № 5. Уплотнение и множественный доступ с частотным разделением каналов .....	27
Список использованных источников .....	31

## ВВЕДЕНИЕ

**Целью** изучения дисциплины является формирование у студентов знаний принципов построения систем связи и сетей передачи информации, основных структурных элементов, основных характеристик систем связи и сетей передачи информации, современных тенденций их развития.

Основные **задачи** дисциплины:

- изучить принципы построения и основные структурные элементы систем связи и сетей передачи информации;
- научить студентов применять полученные знания при разработке систем защиты информации.

При изучении дисциплины акцент делается на выработку у студентов навыков понимания принципов работы систем связи и сетей передачи информации, а также их структурных элементов.

В результате изучения учебной дисциплины студент должен:

**знать:**

- характеристики уровней модели OSI/ISO;
- способы передачи информации по каналам, подверженным ошибкам;
- принципы передачи информации по телефонным сетям общего назначения;
- принципы передачи информации в сетях передачи данных общего пользования;
- методы и оборудование для создания корпоративных сетей;

**уметь:**

- применять полученные знания при разработке систем защиты информации и их интеграции в существующие системы и сети передачи информации;

**владеть:**

- основами методов анализа и проектирования современных систем связи.

Для успешного усвоения дисциплины необходимы знания по математическому анализу, основам радиоэлектроники, компьютерным сетям.

**Лабораторная работа № 1**  
**Работа в среде Matlab.**  
**Структура Signal Processing Toolbox. Генерация сигналов**

**Теоретические сведения**

**1. Общая характеристика пакета Signal Processing**

Signal Processing Toolbox - мощный пакет по анализу, моделированию и проектированию устройств обработки всевозможных сигналов, обеспечению их фильтрации и множества преобразований.

Пакет Signal Processing обеспечивает чрезвычайно обширные возможности по созданию программ обработки сигналов для современных научных и технических приложений. В пакете используется разнообразная техника фильтрации и новейшие алгоритмы спектрального анализа. Пакет содержит модули для разработки новых алгоритмов обработки сигналов, разработки линейных систем и анализа временных рядов. Пакет будет полезен, в частности, в таких областях, как обработка аудио- и видеоинформации, телекоммуникации, геофизика, задачи управления в реальном режиме времени, экономика, финансы и медицина.

Пакет Signal Processing - идеальная оболочка для анализа и обработки сигналов. В нем используются проверенные практикой алгоритмы, выбранные по критериям максимальной эффективности и надежности. Пакет содержит широкий спектр алгоритмов для представления сигналов и линейных моделей. Этот набор позволяет пользователю достаточно гибко подходить к созданию сценария обработки сигналов.

**2. Генерация сигналов в пакете Signal Processing**

Сигналом будем называть данные, упорядоченные относительно некоторого аргумента (например, времени, частоты, пространственной координаты). Если в качестве аргумента выбрано время, то эквивалентными понятиями будут временной процесс, временная реализация.

Для генерации сигналов используются функции среды Matlab, в том числе функции пакета Signal Processing. В пакете Signal Processing реализованы следующие функции генерации сигналов:

Функция	Действие
chirp	Генерирует косинусоиду с переменной частотой
diric	Вычисляет функцию Дирихле
gauspuls	Генерирует синусоиду, модулированную функцией Гаусса
gmonopuls	Генерирует моноимпульс Гаусса
pulstran	Генерирует импульсы

rectpuls	Генерирует аperiodические прямоугольные импульсы
sawtooth	Генерирует пилообразные колебания
sinc	Функция sinc
square	Генерирует прямоугольные импульсы
tripuls	Генерирует аperiodические треугольные импульсы
vco	Управляемый источник напряжений

Подробное описание функций и их параметров можно получить из справочной системы Matlab или используя команду **help <имя функции>**.

### Пример 1

Требуется получить сигнал синусоидальный сигнал с заданной частотой  $f=0,2$  Гц различными частотами дискретизации  $f_d=1$  Гц, 3 Гц, 10 Гц. Длина сигнала 20 с.

Так как работа среды Matlab ориентирована на матричное представление сигналов, то зададим сначала три вектора времени  $t$ , которые будут соответствовать различным частотам снятия данных (частотам дискретизации).

```
fs1=1; fs2=3; fs3=10; %fs – sampling frequency – частота дискретизации
T=20; f=0.2; % T – длина сигнала ; f- частота сигнала
t1=0:1/fs1:T;
t2=0:1/fs2:T;
t3=0:1/fs3:T;
```

Для генерации синусоидального сигнала (вычисления функции  $\sin$  в заданных вектором  $t_i$  точках используется функция Matlab –  $\sin$ . Уточним, требуемые данной функцией, аргументы:

Далее зададим вектор сигналов:

```
y1=sin(2*pi*f*t1);
y2=sin(2*pi*f*t2);
y3=sin(2*pi*f*t3);
```

Для визуализации сигналов воспользуемся функцией plot:

```
plot(t1,y1,'-ro',t2,y2,'-g>',t3,y3,'-b')
grid on % Отображение сетки
xlabel('Time'); % Подпись оси X
ylabel('Amplitude'); % Подпись оси
title('SIGNAL'); % Заголовок рисунка
```

Результат работы изображен на рис.1.

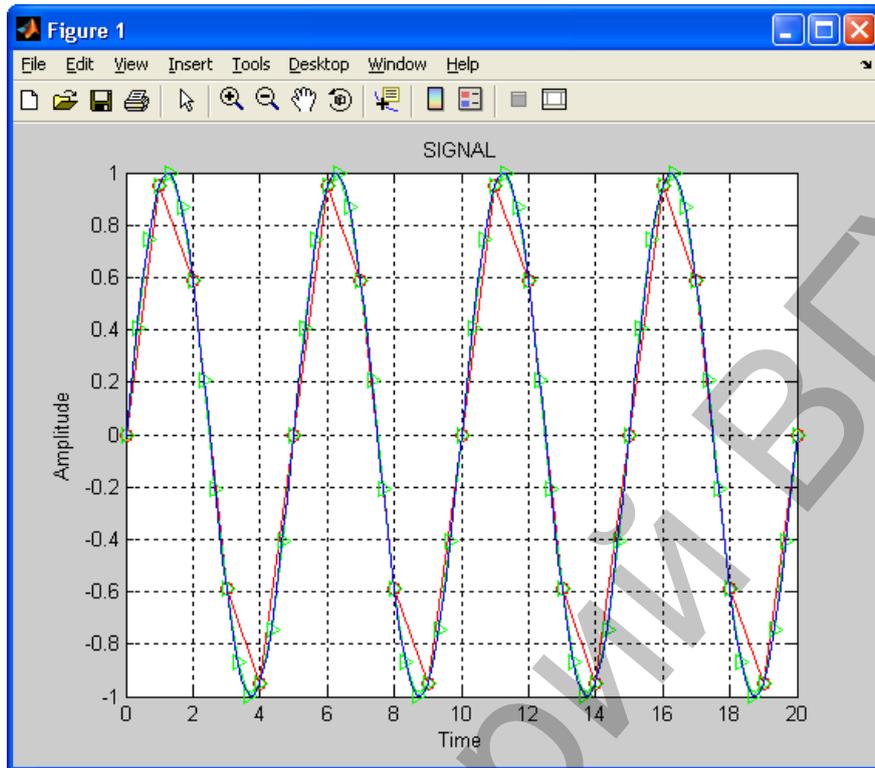


Рисунок 1. Графики сигналов

Все вышеизложенные операции могут быть оформлены в виде m-файла в редакторе m-файлов (рис. 2).

```
Editor - C:\Documents and Settings\Admin\Мои документы\MATLAB\Untitled6.m
File Edit Text Go Cell Tools Debug Desktop Window Help
1 - fs1=1; fs2=3; fs3=10; %fs - sampling frequency - частота дискретиза
2 - T=20; f=0.2; % T - длина сигнала ; f- частота сигнала
3 - t1=0:1/fs1:T;
4 - t2=0:1/fs2:T;
5 - t3=0:1/fs3:T;
6 - y1=sin(2*pi*f*t1);
7 - y2=sin(2*pi*f*t2);
8 - y3=sin(2*pi*f*t3);
9 - plot(t1,y1,'-ro',t2,y2,'-g>',t3,y3,'-b')
10 - grid on % Отображение сетки
11 - xlabel('Time'); % Подпись оси X
12 - ylabel('Amplitude'); % Подпись оси
13 - title('SIGNAL'); % Заголовок рисунка
script Ln 13 Col 37 OVR
```

Рисунок 2. Редактор m-файлов

## Пример 2

Требуется получить функцию MATLAB генерирующую сигнал длительностью 10 с, снятый с шагом  $1/f$  и представляющий собой функцию:

$$y(x) = \begin{cases} \sin(20\pi t), & t \in [0,3]c \\ 2 + \sin(20\pi t), & t \in [3,6]c \\ \sin(10\pi t), & t \in [6,10]c \end{cases},$$

где параметр  $f$  - есть аргумент функции.

Один из возможных вариантов решения данной задачи приведен ниже.

```
f=128;  
t=0:1/f:10;  
t1=0:1/f:3;  
A(1:length(t1))=sin(7*pi*t1);  
t2=(3+1/f):1/f:6;  
A(length(t1)+1:length(t1)+length(t2))=2+sin(7*pi*t2);  
t3=(6+1/f):1/f:10;  
A((length(t1)+length(t2)+1):(length(t1)+length(t2)+length(t3)))=sin(  
2*pi*t3);  
plot(t,A);  
grid on;
```

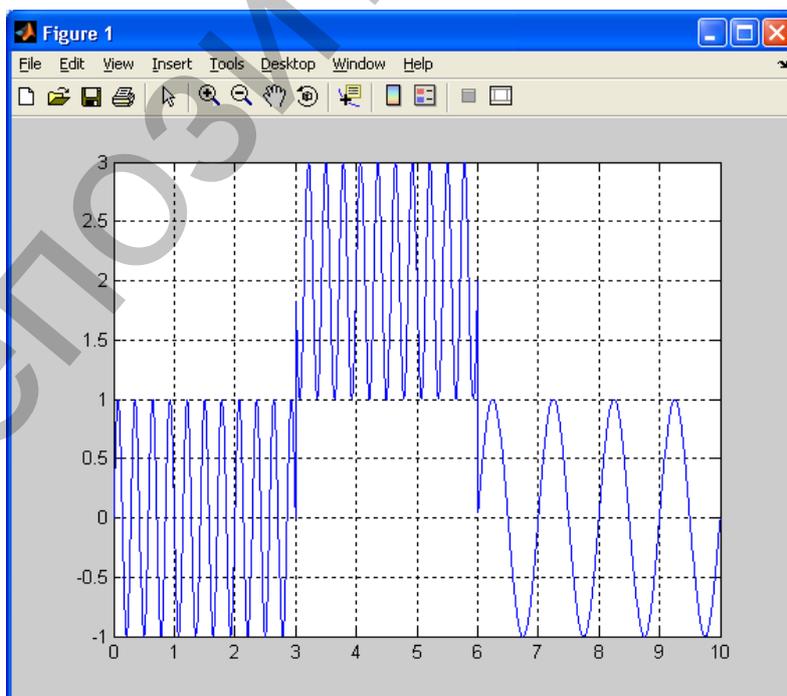


Рисунок 3. График искомого сигнала

### Пример 3

В данном примере генерируется последовательность асимметричных пилообразных импульсов длительностью 0,1 с, следующих друг за другом с частотой 3 Гц. Длительность генерируемого сигнала равна 1 с, частота дискретизации составляет 1 кГц:

```
t = 0:1/1e3:1; % интервал 1с дискретизирован с частотой 1 кГц  
d = 0:1/3:1; % вектор задержек - частота импульсов 3 Гц
```

```
y = pulstran(t,d,'tripuls',0.1,-1); plot(t,y)
```

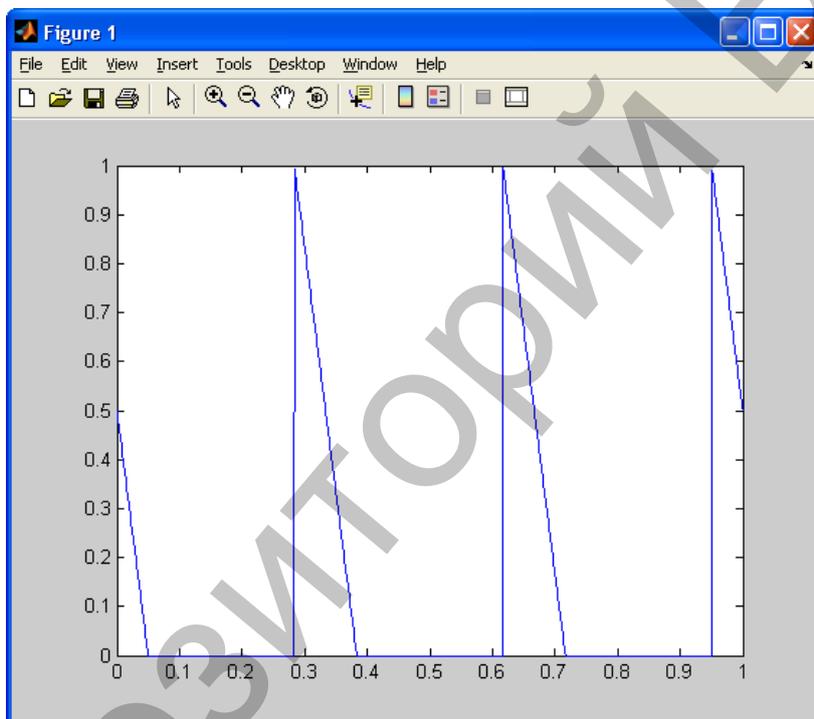


Рисунок 4. График искомого сигнала

### Задание

1. Выполнить генерацию сигналов:

Описание сигнала	Параметры					
	A, Гц	B, Гц	C, Гц	D, Гц	Длительность, с	Частота дискретизации, Гц
Гармонический сигнал с частотой A	10				10	50
	20				20	60
	30				30	70

с нормально распределенным шумом (randn)						
Синусоида с меняющейся частотой от А до В по законам С и D	5	100	Лин.	Квадр.	10	200
	10	120	Лин.	Эксп.	20	300
	1	10	Эксп.	Квадр.	30	50
Синусоида с частотой меняющейся скачкообразно: первая частота А, далее В, С и D	10	20	30	40	10	50
	10	20	10	20	20	50
	100	10	50	5	30	200

2. С помощью функции `rectpuls` создайте одиночный двухполярный импульс, изображенный на рисунке 5.

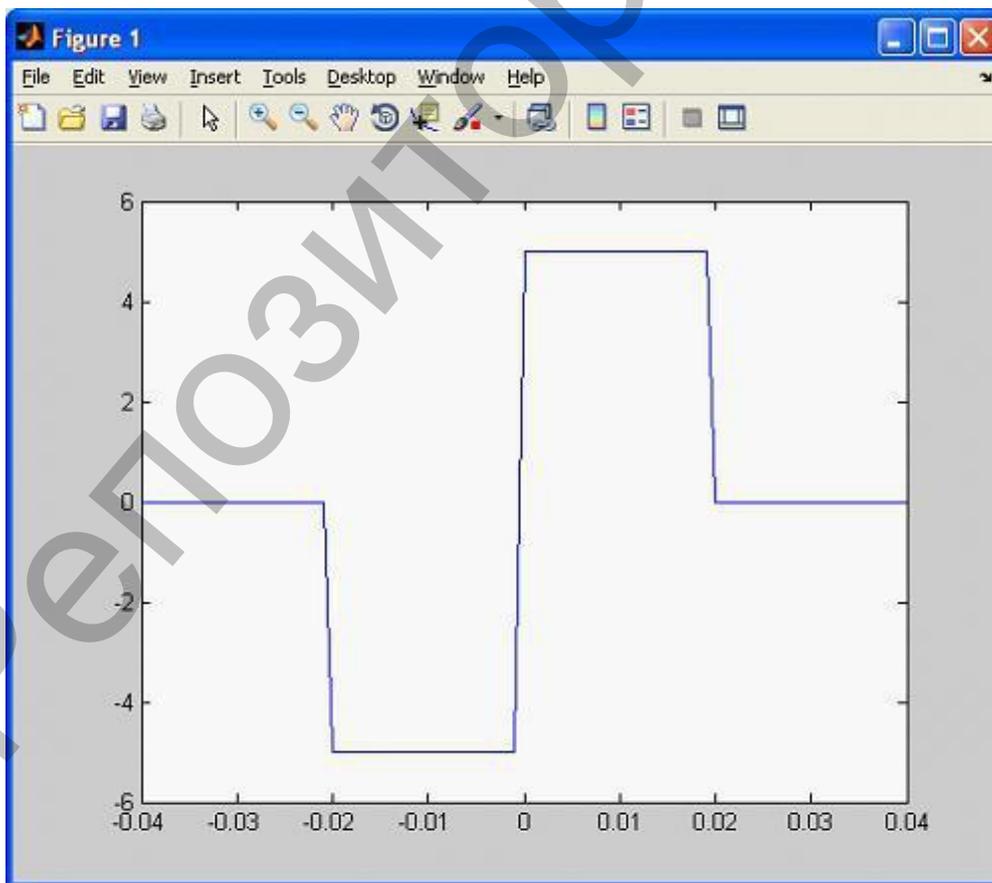


Рисунок 5. Одиночный двухполярный импульс.

3. С помощью функции `trimpuls` создайте одиночный трапецидальный импульс, изображенный на рисунке 6:

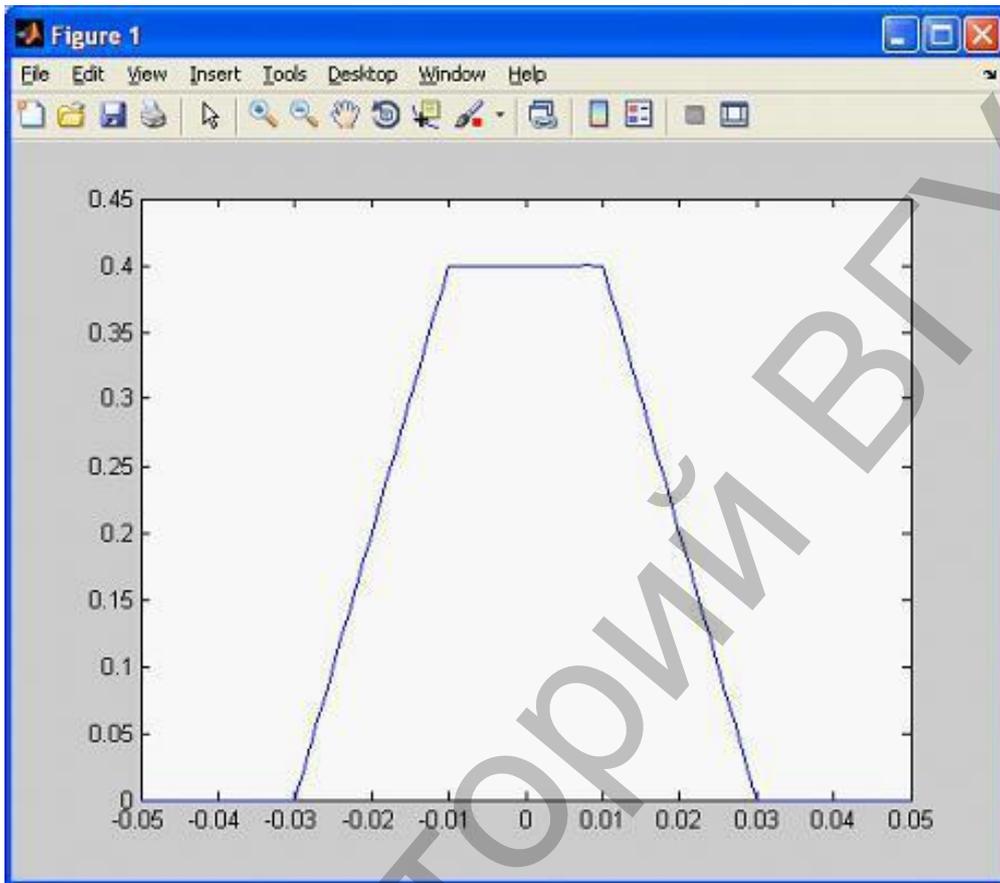


Рисунок 6. Одиночный трапецидальный импульс.

**Лабораторная работа № 2**  
**Исследование свойств преобразования Фурье**  
**дискретных сигналов**

**Теоретические сведения**

**1. Преобразование Фурье**

Анализ дискретного сигнала в частотной области осуществляется с помощью дискретного преобразования Фурье (ДПФ):

- прямого:

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-i \frac{2\pi}{N} nk}, k = 0, 1, \dots, N-1$$

- обратного:

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{i \frac{2\pi}{N} nk}, n = 0, 1, \dots, N-1$$

где  $N$  – количество отсчетов входного сигнала и отсчетов ДПФ  $X(k)$  на периоде.

Спектральная плотность сигнала вычисляется как:  $S(k) = |X(k)|^2$ .

Для повышения быстродействия коэффициенты ДПФ  $X(k)$  рассчитываются с помощью алгоритма БПФ Кули–Тьюки с использованием функции `fft()`, формат которой имеет вид:

$$X = \text{fft}(x)$$

или

$$X = \text{fft}(x, m)$$

где:

$x$  – вектор отсчетов входного сигнала  $x(n)$ ;

$X$  – вектор отсчетов ДПФ  $X(k)$ ;

$m$  – количество отсчетов сигнала  $x(n)$  по которым вычисляется ДПФ.

Если  $m < N$ , к вектору  $x$  добавляются нулевые элементы, если  $m > N$ , элементы вектора  $x$ , начиная с  $n > m$  при вычислении ДПФ не учитываются.

**Пример 1**

Основное назначение преобразования Фурье - выделить частоты регулярных составляющих сигнала, зашумленного помехами. Рассмотрим данные, поступающие с частотой 1000 Гц. Сформируем сигнал, содержащий регулярные составляющие с частотами 50 Гц и 120 Гц и случайную аддитивную компоненту с нулевым средним. Длина сигнала 0,6 с.

```

fd = 1000;
f1 = 50;
f2 = 120;
Tc = 0.6;
t = 0:1/fd:Tc;
x = sin(2 * pi * f1 * t) + sin(2 * pi * f2 * t);
y = x + 2 * randn(size(t));
figure(1);
plot(y(1:50)), grid

```

На рисунке 1 показан этот сигнал. Глядя на него, трудно сказать, каковы частоты его регулярных составляющих. Реализуя одномерное преобразование Фурье этого сигнала на основе 512 точек и построив график спектральной плотности, можно выделить две частоты, на которых амплитуда спектра максимальна. Это частоты 120 и 50 Гц.

```

figure(3);
Y = fft(y, fd);
Pyy = Y.*conj(Y);
figure(3), plot(Pyy(1:500)), grid

```

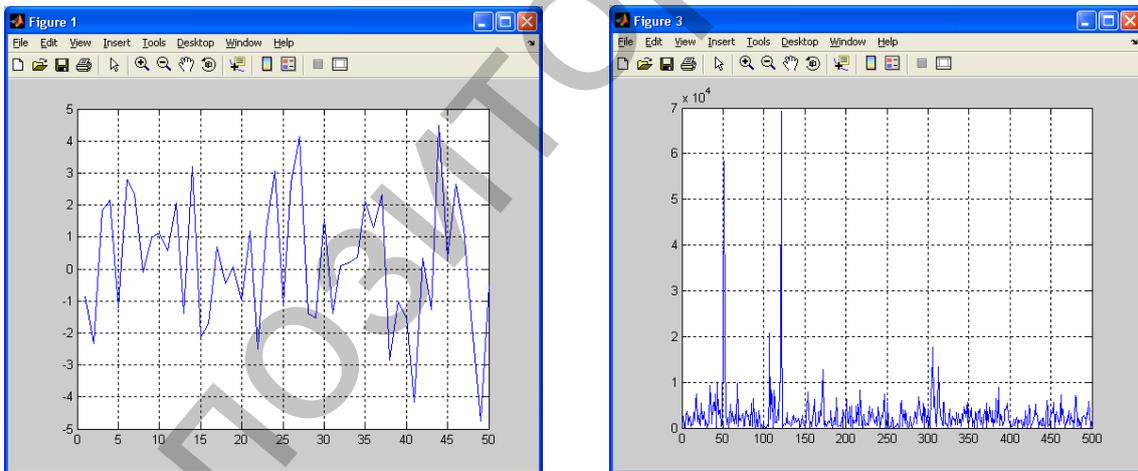


Рисунок 1. Сигнал и спектральная плотность

Спектральную плотность можно высчитать, также, следующим образом. Отсчеты ДПФ Y в общем случае – комплексные числа, поэтому модуль отсчетов ДПФ определяется как  $\text{abs}(X)$ .

```

figure(2);
Y1 = fft(y, fd);
Y1 = abs(Y1);
plot(Y1(1:500).^2);

```

## 2. Спектр сигнала, ограниченного во времени

Исследователь никогда не имеет дела с сигналом в полной его реализации от  $-\infty$  до  $+\infty$ . Сигналы рассматриваются в каком-то временном промежутке. Известно, что спектр при ограничении сигнала во времени расширяется.

## 3. Эффект подмены частот

Дискретный сигнал получается из аналогового операцией дискретизации – взятием отсчетов (измерением) через интервал времени  $T$ .

При этой операции представляется возможной потеря информации, заключенной в значениях сигнала в интервалах между отсчетами. Условия, при которых осуществимо восстановление аналогового сигнала по полученному из него цифровому, то есть сохранение всей исходно содержащейся в сигнале информации, выражаются теоремой Котельникова.

Для этого требуется, чтобы полоса частот входного сигнала была бы не менее чем вдвое уже, чем частота дискретизации, то есть  $f_c = 1/2f_d$ . Нередко приводят частную ее формулировку, верную для сигналов, чья полоса частот начинается с нулевой частоты – «чтобы не присутствовали частоты большие, нежели половина частоты дискретизации». Если же такие частоты имеются, возникает эффект маскировки (подмены) частот.

## 4. Задание

1. Вычислить спектральную мощность сигналов (с шумом):

Вар	Сигнал	Длительность	$f_1$ Гц	$f_2$ Гц	$f_0$ Гц
1	$x(n) = 0,7 \sin(2\pi f_1 n) + 0,5 \sin(2\pi f_2 n)$	$T_c = 1$	200	300	2000
2			300	500	3000
3			330	488	1700
4			460	566	1900
5			120	245	1322
6			23	45	580
7			35	67	690

2. Найти спектр мощности сигнала, имеющего различную длину. Сравнить спектры, сравнить полученный результат с теоретическим.

3. Продемонстрировать эффект подмены частот. Для этого вычислить спектр сигнала с частотой дискретизации удовлетворяющей теореме Котельникова, и не удовлетворяющей. Сравнить результаты.

## Лабораторная работа № 3

### Цифровая модуляция сигналов

#### Теоретические сведения

В настоящее время все большая часть информации, передаваемой по разнообразным каналам связи, существует в цифровом виде. Это означает, что передаче подлежит не непрерывный (аналоговый) модулирующий сигнал, а последовательность целых чисел  $n_0, n_1, n_2, \dots$ , которые могут принимать значения из некоторого фиксированного конечного множества. Эти числа, называемые *символами* (symbol), поступают от источника информации с периодом  $T$ , а частота, соответствующая этому периоду, называется *символьной скоростью* (symbol rate):  $f_T = 1/T$ .

Часто используемым на практике вариантом является *двоичная* (binary) последовательность символов, когда каждое из чисел  $n_i$  может принимать одно из двух значений – 0 или 1.

Типичный подход при осуществлении передачи дискретной последовательности символов состоит в следующем. Каждому из возможных значений символа сопоставляется некоторый набор параметров несущего колебания. Эти параметры поддерживаются постоянными в течение интервала  $T$ , то есть до прихода следующего символа. Фактически это означает преобразование последовательности чисел  $\{n_k\}$  в ступенчатый сигнал  $s_n(t)$  с использованием кусочно-постоянной интерполяции:

$$s_n(t) = f(n_k), kT \leq t < (k + 1)T.$$

Здесь  $f$  – некоторая функция преобразования. Полученный сигнал  $s_n(t)$  далее используется в качестве модулирующего сигнала обычным способом.

Такой способ модуляции, когда параметры несущего колебания меняются скачкообразно, называется *манипуляцией* (keying). В зависимости от того, какие именно параметры изменяются, различают амплитудную (АМн), фазовую (ФМн), частотную (ЧМн).

#### Амплитудная манипуляция

Как будет показано далее, амплитудная манипуляция (АМн; английский термин – amplitude shift keying, ASK), при которой скачкообразно меняется *амплитуда* несущего колебания, является частным случаем квадратурной манипуляции. Поэтому здесь мы только построим в качестве примера график АМн-сигнала и скажем несколько слов о демодуляции сигналов данного типа.

Демодуляция АМн-сигнала может выполняться теми же методами, что и в случае квадратурной манипуляции (путем умножения на

несущее колебание). Однако наличие всего лишь двух возможных значений начальной фазы несущей, отличающихся друг от друга на  $180^\circ$ , делает возможной реализацию автоматической подстройки начальной фазы с помощью петли ФАПЧ. Этот режим демодуляции реализуется функциями `ddemod` и `ddemodce` при указании вида манипуляции `'ask/costas'`.

### Пример 1

Амплитудная манипуляция осуществляется функциями `dmod` (формируется вещественный выходной сигнал) и `dmodce` (формируется комплексная огибающая) пакета `Communications` при указании в них параметра типа модуляции `'ask'`. Следующий за ним параметр `M` указывает количество используемых уровней манипуляции. Символы, подлежащие передаче, должны принимать целочисленные значения, лежащие в диапазоне  $0 \dots M-1$ . Символу `0` соответствует значение амплитуды, равное  $-1$ , а символу `M-1` – значение амплитуды, равное  $1$ . Остальные уровни равномерно распределены между этими значениями. Таким образом, строго говоря, в данном случае может меняться не только амплитуда, но и фаза несущего колебания (отрицательные амплитудные множители соответствуют изменению фазы на  $180^\circ$ ).

В качестве примера построим график сигнала, содержащего все возможные символы при 8-позиционной АМн:

```
M = 8; % количество уровней манипуляции
sy = 0:M-1; % передаваемые символы
Fd = 1; % символьная скорость
Fc = 4; % несущая частота
FsFd = 40; % отношение Fs/Fd
Fs = Fd * FsFd; % частота дискретизации
% формируем АМн-сигнал
[s_ask, t] = dmod(sy, Fc, Fd, Fs, 'ask', M);
plot(t, s_ask)
grid on
```

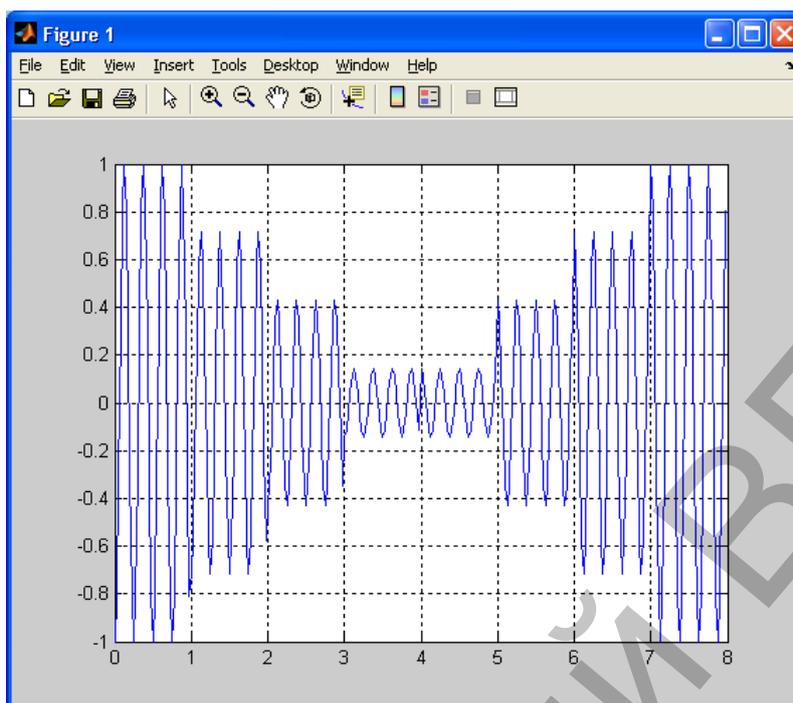


Рисунок 1. График искомого сигнала примера 1

На графике хорошо виден бросок фазы в середине сигнала. Кроме того, можно заметить, что фазы посылок в первой и второй половинах сигнала, имеющих одинаковые амплитуды, отличаются на  $180^\circ$ .

### **Фазовая манипуляция**

Фазовая манипуляция (ФМн; английский термин – phase shift keying, PSK), при которой скачкообразно меняется фаза несущего колебания, тоже является частным случаем квадратурной манипуляции.

На практике фазовая манипуляция используется при небольшом числе возможных значений начальной фазы – как правило, 2, 4 или 8. Кроме того, при приеме сигнала сложно измерить абсолютное значение начальной фазы; значительно проще определить относительный фазовый сдвиг между двумя соседними символами. Поэтому обычно используется фазоразностная манипуляция (синонимы – дифференциальная фазовая манипуляция, относительная фазовая манипуляция; английский термин – differential phase shift keying, DPSK).

Демодуляция фазовой манипуляции может выполняться тем же методом, что и в случае квадратурной манипуляции (путем умножения на несущее колебание). Применительно к ФМн данный метод демодуляции часто называется корреляционным.

### **Пример 2**

Фазовая манипуляция осуществляется функциями `dmod` (формируется вещественный выходной сигнал) и `dmodse` (формируется ком-

плексная огибающая) пакета Communications при указании в них параметра типа модуляции 'psk'. Следующий за ним параметр  $M$  указывает количество используемых градаций начальной фазы. Символы, подлежащие передаче, должны принимать целочисленные значения, лежащие в диапазоне  $0 \dots M-1$ . Символу  $k$  соответствует значение начальной фазы, равное  $2\pi k/M$  радиан, или  $360k/M$  градусов.

В качестве примера построим график сигнала, содержащего все возможные символы при 4-позиционной ФМн:

```
M = 4; % количество позиций манипуляции
sy = 0:M-1; % передаваемые символы
Fd = 1; % символьная скорость
Fc = 4; % несущая частота
FsFd = 40; % отношение Fs/Fd
Fs = Fd * FsFd; % частота дискретизации
% формируем ФМн-сигнал
[s_psk, t] = dmod(sy, Fc, Fd, Fs, 'psk', M);
plot(t, s_psk)
```

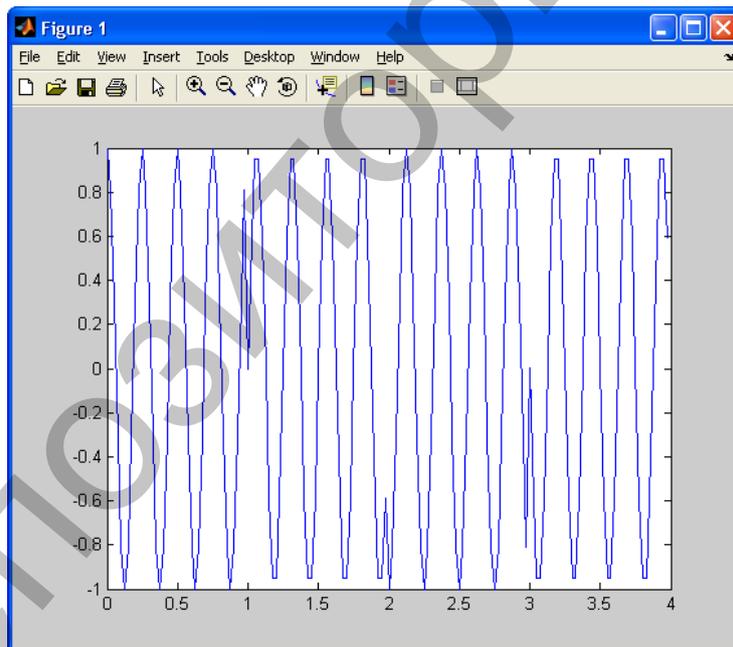


Рисунок 2. График искомого сигнала примера 2

На графике видны скачки фазы на  $90^\circ$ , происходящие при переходе от одного символа к другому.

### Частотная манипуляция

При частотной манипуляции (ЧМн; английский термин – frequency shift keying, FSK) каждому возможному значению переда-

ваемого символа сопоставляется своя частота. В течение каждого символьного интервала передается гармоническое колебание с частотой, соответствующей текущему символу. При этом возможны различные варианты, различающиеся выбором начальной фазы отдельных синусоидальных посылок.

Первый способ – когда все посылки, соответствующие одному закону передаваемого символа, имеют одинаковую начальную фазу, то есть являются идентичными. При этом можно заранее сформировать наборы отсчетов для всех возможных дискретных символов. Тогда осуществление частотной манипуляции сводится к последовательной передаче заранее рассчитанных последовательностей отсчетов, соответствующих поступающим символам. Однако если используемые частоты манипуляции не кратны символьной скорости, сформированный таким образом ЧМн-сигнал будет содержать разрывы (скачки) на стыках символов. Вследствие этого спектр сигнала будет иметь всплески на частотах, кратных символьной скорости.

Второй способ – непрерывная генерация колебаний всех необходимых частот и осуществление переключения между этими сигналами в соответствии с поступающими символами. Данный метод также не гарантирует отсутствие скачков на стыках символов, но вследствие того, что начальные фазы посылок меняются от символа к символу, скачки возникают не на всех стыках, и их величина оказывается различной. В результате возникающие из-за скачков всплески спектра в данном случае выражены слабее. Именно этот вариант формирования ЧМн-сигнала используется в функциях `dmod` и `dmodce` пакета `Communications`.

Наконец, третий способ – когда поступающие для передачи символы управляют скоростью линейного нарастания текущей фазы, а частотно-манипулированный сигнал формируется путем вычисления косинуса этой текущей фазы. При этом фазовая функция, а значит, и сам ЧМн-сигнал оказываются непрерывными (не имеющими скачков). Данный способ сложнее в реализации, но он дает наиболее компактный спектр сигнала. ЧМн-сигнал, полученный таким образом, называется частотно-манипулированным сигналом с непрерывной фазовой функцией (`continuous phase frequency shift keying – CPFSK`).

### **Пример 3**

Частотная манипуляция осуществляется функциями `dmod` (формируется вещественный выходной сигнал) и `dmodce` (формируется комплексная огибающая) пакета `Communications` при указании в них параметра типа модуляции `'fsk'`. Следующие за ним параметры `M` и `tone` указывают соответственно количество используемых частот манипуляции и расстояние между соседними частотами (по умолчанию

значение параметра  $\text{tone}$  равно символьной скорости  $F_d$ ). Символы, подлежащие передаче, должны принимать целочисленные значения, лежащие в диапазоне  $0 \dots M-1$ . Символу  $k$  соответствует смещение частоты (относительно несущей частоты  $F_c$ ), равное  $\text{tone} \cdot (1 - M + 2 \cdot k) / 2$ .

В качестве примера сформируем 2-позиционный (бинарный) ЧМн-сигнал, в котором возможным значениям символов 0 и 1 соответствуют частоты 800 и 1600 Гц. Символьная скорость будет равна 400 символам в секунду, а частота дискретизации – 16 кГц:

```
bits = [0 1 0 0 1]; % цифровое сообщение
N = length(bits); % длина сообщения
Fd = 400; % символьная скорость
FsFd = 40; % отношение Fs/Fd
Fs = Fd * FsFd; % частота дискретизации
f0 = 800; % частота манипуляции для символа "0"
f1 = 1600; % частоты манипуляции для символа "1"
Fc = (f0 + f1)/2; % несущая частота
tone = f1 - f0; % разнос частот
% формируем ЧМн-сигнал
[s_fsk, t] = dmod(bits, Fc, Fd, Fs, 'fsk', 2, tone);
td = t * Fd; % время для графика – в символах
plot(td, s_fsk)
xlabel('Symbols')
ylabel('s_{FSK}')
ylim([-1.1 1.1])
```

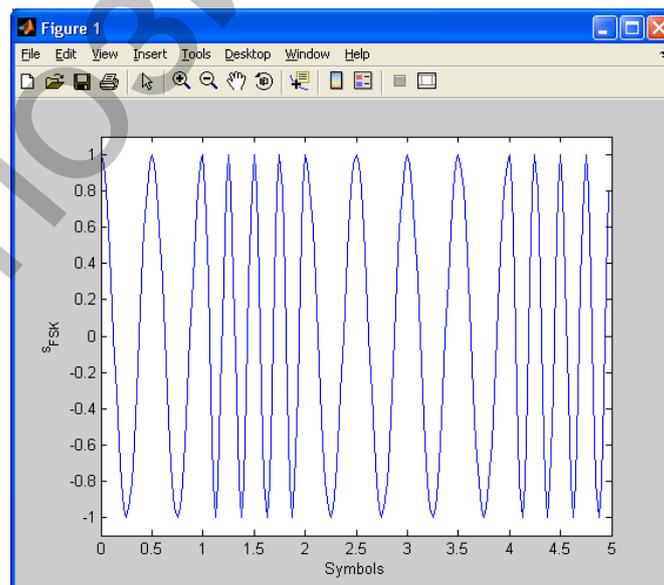


Рисунок 3. График искомого сигнала примера 3

На графике хорошо заметно двукратное изменение частоты сигнала при смене значения передаваемого бита. В данном примере на длительности символа укладывается два периода колебания при передаче нулевого бита и четыре периода – при передаче единичного бита.

### Демодуляция ЧМн-сигнала

Прием ЧМн-сигнала, как правило, осуществляется корреляционным методом. При этом корреляционный прием может быть когерентным или некогерентным. Когерентный метод может использоваться, если известны начальные фазы посылок. Сущность его состоит в вычислении взаимной корреляции между принимаемым сигналом и колебаниями-образцами (опорными сигналами), представляющими собой гармонические колебания с используемыми для манипуляции частотами.

Когерентный вариант демодуляции используется функциями `ddemod` и `ddemodsc` по умолчанию. Для использования некогерентного варианта необходимо при вызове этих функций указать метод манипуляции в виде `'fsk/noncoherence'`.

### Пример 4

В качестве примера оценим помехоустойчивость частотной манипуляции при когерентной и некогерентной демодуляции, смоделировав случайный информационный сигнал, сформировав соответствующий ЧМн-сигнал, добавив к нему шум и осуществив когерентную и некогерентную демодуляцию зашумленного сигнала. Повторив эту процедуру при различных отношениях сигнал/шум, получим графики помехоустойчивости. Параметры ЧМн-сигнала выберем соответствующими нижнему частотному каналу рекомендации ITU-T V.21 (этот протокол используется модемами в качестве “наименьшего общего знаменателя” на самом раннем этапе вхождения в связь): символьная скорость 300 символов/с, манипуляция двоичная, символу “0” соответствует частота манипуляции 1180 Гц, символу “1” – 980 Гц. Частоту дискретизации (напомним, что при использовании функций пакета `Communications` она должна быть кратна символьной скорости) выберем равной 9600 Гц. Вот соответствующий код:

```
N = 10000; % число передаваемых бит
x = randint(N, 1); % цифровое сообщение
M = 2; % двоичная манипуляция
Fd = 300; % символьная скорость
Fs = 9600; % частота дискретизации
f0 = 1180; % частота “нуля”
f1 = 980; % частота “единицы”
```

```

Fc = (f0 + f1) / 2; % средняя частота
tone = f1 - f0; % разнос частот
s = dmod(x, Fc, Fd, Fs, 'fsk', M, tone); % манипулированный сигнал
snr = -10:10; % вектор отношений С/Ш (в децибелах)
for k = 1:length(snr)
sn = awgn(s, snr(k), 'measured'); % добавляем шум
% когерентная демодуляция
y_c = ddemod(sn, Fc, Fd, Fs, 'fsk', M, tone);
% некогерентная демодуляция
y_nc = ddemod(sn, Fc, Fd, Fs, 'fsk/noncoherence', M, tone);
% расчет вероятностей ошибок
[n_err, er_c(k)] = symerr(x, y_c);
[n_err, er_nc(k)] = symerr(x, y_nc);
end
% вывод графика
semilogy(snr, er_c, snr, er_nc)

```

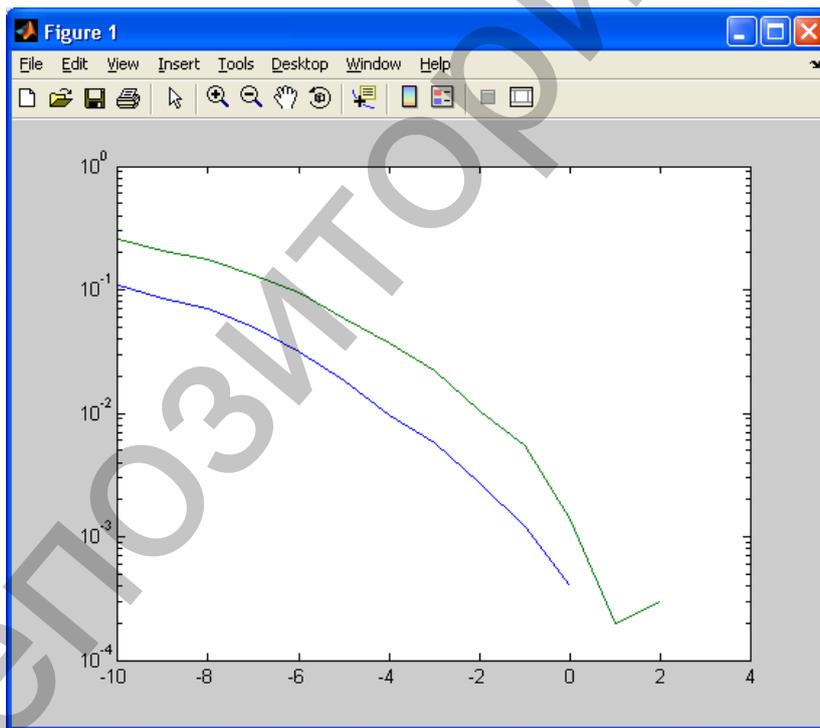


Рисунок 4. График искомого сигнала примера 4

Приведенный график показывает зависимость вероятности ошибки от отношения сигнал/шум (в децибелах). Синяя кривая соответствует когерентной демодуляции, зеленая – некогерентной. Видно, что проигрыш некогерентного варианта когерентному составляет от 1 до 3 дБ.

При моделировании демодуляции сигнала с отношением сигнал/шум, равным 2 дБ и более, ошибок приема не возникло. Поэтому при выводе графика вероятности ошибок с использованием логарифмического масштаба по вертикали эти точки оказались отброшенными.

Из графика также видно, что помехоустойчивость данного вида манипуляции очень высока – даже при равенстве средних мощностей сигнала и шума (отношение сигнал/шум 0 дБ) вероятность ошибки составляет примерно  $2 \cdot 10^{-4}$  для когерентного варианта и примерно  $1,5 \cdot 10^{-3}$  – для некогерентного. Платой за это в данном случае является крайне низкая скорость передачи данных – всего лишь 300 бит/с.

### Задание

1. Выполнить приведенные примеры.
2. Выполнить модуляцию сигнала всеми тремя способами, передать слово «Hello!» в ASCII кодировке:

	Количество уровней манипуляции	Символьная скорость, Бод	Частота дискретизации, Гц	Частота несущей, Гц
1.	2	10	100	30
2.	3	20	200	60
3.	4	30	300	90
4.	5	40	400	120

3. Рассчитать спектры модулированных сигналов, вывести графики, определить практическую ширину спектра модулированного сигнала.
4. Рассчитать девиацию частоты для всех видов модуляции.

## Лабораторная работа № 4 Канальное кодирование сигналов

### 1. Кодирование данных с использованием блочных кодов с помощью функции encode

Функция encode кодирует сообщения, используя один из следующих помехоустойчивых (корректирующих) блочных кодов:

- Линейные блочные коды общего вида
- Циклические коды
- Коды Боуза–Чоудхури–Хоквингема (БЧХ)
- Коды Хэмминга
- Коды Рида–Соломона

### 2. Общая информация о синтаксисе функции:

Синтаксис вызова функции encode можно посмотреть на адресу <http://matlab.exponenta.ru/communication/book2/4/encode.php>.

Для всех кодов параметры функции выбираются следующим образом:

$m$  – число, большее 3.

$n = 2^m - 1$  - длина кодового слова (кодовой комбинации),

$k = n - m$  - длина блока сообщения (информационной комбинации).

### 3. Пример кодирования последовательности кодом Хэмминга

Установить параметры кодирования

```
n = 15;           % Code length
k = 11;           % Message length
```

Создать двоичное сообщение, имеющее длину  $k$ .

```
data = randi([0 1],k,1);
```

Закодировать сообщение.

```
encData = encode(data,n,k,'hamming/binary');
```

Сделать ошибку в 4-м бите закодированной последовательности.

```
encData(4) = ~encData(4);
```

Расшифровать поврежденную последовательность. Обратите внимание, что декодер правильно восстановил сообщение.

```
decData = decode(encData,n,k,'hamming/binary');
numerr = biterr(data,decData)
```

#### **4. Пример кодирования последовательности линейным блочным кодом**

Установить параметры кодирования

```
n = 7;          % Code length
k = 3;          % Message length
```

Создать двоичное сообщение, имеющее длину  $k$ .

```
data = randi([0 1],k,1);
```

Создать матрицу генератор:

```
pol = cyclpoly(n,k);
parmat = cyclgen(n,pol);
genmat = gen2par(parmat);
```

Закодировать последовательность сообщений с помощью порождающей матрицы.

```
encData = encode(data,n,k,'linear/binary',genmat);
```

Ввести ошибку в 3-й бит закодированной последовательности.

```
encData(3) = ~encData(3);
```

Расшифровать поврежденную последовательность. Обратите внимание, что декодер правильно восстановит сообщение.

```
decData = decode(encData,n,k,'linear/binary',genmat);
numerr = biterr(data,decData)
```

#### **5. Пример кодирования последовательности циклическим кодом**

Установить параметры кода.

```
n = 15;          % Code length
k = 5;           % Message length
```

Создать двоичное сообщение, имеющее длину  $k$ .

```
data = randi([0 1],k,1);
```

Создать матрицу генератор

```
gpol = cyclpoly(n,k);  
parmat = cyclgen(n,gpol);
```

Создать синдром декодирования, с помощью матрицы контроля четности.

```
trt = syndtable(parmat);
```

Закодировать последовательность сообщений с помощью порождающей матрицы.

```
encData = encode(data,n,k,'cyclic/binary',gpol);
```

Внести ошибки в 4-й и 7-й биты закодированной последовательности.

```
encData(4) = ~encData(4);  
encData(7) = ~encData(7);
```

Расшифровать поврежденную последовательность. Обратите внимание, что декодер правильно восстановит сообщение.

```
decData = decode(encData,n,k,'cyclic/binary',gpol,trt);  
numerr = biterr(data,decData)
```

### Задание

1. Выполнить кодирование сигналов с помощью функции `encode`, передать слово «Hello!» в ASCII кодировке (бинарный код) используя один из следующих помехоустойчивых (корректирующих) блочных кодов:
  - Линейные блочные коды общего вида
  - Циклические коды
  - Коды Хэмминга
  - Коды Боуза–Чоудхури–Хоквингема (БЧХ)
  - Коды Рида–Соломона
2. Для одного из кодов выполнить модуляцию с шумом и демодуляцию сигнала из лабораторной работы 3. Определить вероятности ошибки для закодированного сигнала и исходного.

## Лабораторная работа № 5

### Уплотнение и множественный доступ с частотным разделением каналов

#### Теоретические сведения

Распределение по каналам, характеризующееся ортогональными волнами, для которых выполняется условие (1), называют уплотнением с частотным разделением (frequency-division multiplexing – FDM) или множественным доступом с частотным разделением (frequency-division multiple access – FDMA).

$$\int_{-\infty}^{\infty} x_i(f)x_j(f)df = \begin{cases} K & \text{при } i = j \\ 0 & \text{при } i \neq j \end{cases} \quad (1)$$

Ресурс связи может одновременно содержать несколько сигналов, разнесенных в спектре. Первый частотный диапазон содержит сигналы, которые используют промежуток частот между  $f_0$  и  $f_1$  второй – между  $f_2$  и  $f_3$  и т.д. Области спектра, находящиеся между используемыми диапазонами, называют защитными полосами частот. Защитные полосы выполняют роль буфера, что позволяет снизить интерференцию между соседними (по частоте) каналами.



Рисунок 1. Управление частотным разделением

Может возникнуть вопрос: как преобразовать немодулированный сигнал так, чтобы он использовал более высокий диапазон частот? Ответ: при помощи наложения или смешивания (модуляции) информационного сигнала и синусоидального сигнала фиксированной частоты.

На рисунке 2 а показано модулирование типичного голосового телефонного сигнала  $x(t)$  (частоты немодулированного сигнала принадлежат диапазону 300-3400 Гц) синусоидальным сигналом с частотой

20 кГц. Двусторонний спектр немодулированного сигнала,  $|X(f)|$ , показан на рисунке.

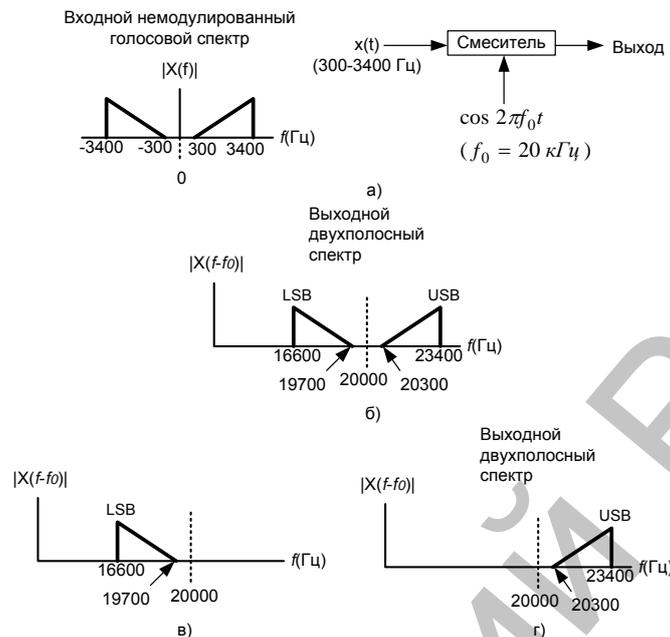


Рисунок 2. Смешивание сигналов: а) процесс смешивания, б) выходной спектр смесителя, в) нижняя боковая полоса, г) верхняя боковая полоса

### Используемые функции

$$y = \text{ssbmod}(x, F_c, F_s)$$

Использует информационный входной сигнал  $x$  для однополосной амплитудной модуляции с нижней боковой полосой. Частота несущего колебания  $F_c$  (Гц). Несущее колебание и модулирующий сигнал  $x$  имеют одинаковую частоту дискретизации  $F_s$  (Гц). Модулированный сигнал имеет нулевую начальную фазу.

$$z = \text{ssbdemod}(y, F_c, F_s)$$

Демодулирует сигнал  $y$ , модулированный с использованием однополосной АМ при частоте несущего колебания  $F_c$  (Гц). Несущее колебание и модулированный сигнал  $y$  имеют одинаковую частоту дискретизации  $F_s$  (Гц). Предполагается, что модулированный сигнал  $y$  имеет нулевую начальную фазу; использованная при модуляции боковая полоса не имеет значения. В процессе демодуляции используется фильтр нижних частот Баттерворта, рассчитываемый как  $[\text{num}, \text{den}] = \text{butter}(5, F_c * 2 / F_s)$ .

**Замечание.** Входные параметры  $F_c$  и  $F_s$  должны удовлетворять условию  $F_s > 2(F_c + BW)$ , где  $BW$  - ширина спектра информационного (модулирующего) сигнала.

**Y = wavread(FILE)** – считывает файл типа WAVE с именем FILE и возвращает данные в массиве Y;

**[Y, FS, BITS] = wavread(FILE)** – считывает файл типа WAVE с именем FILE и возвращает массив данных Y, частоту дискретизации FS (в герцах) и разрядность BITS кодирования звука (в битах);

**wavwrite(Y, WAVEFILE)** – записывает файл типа WAVE под именем WAVEFILE. Данные по каждому каналу в случае стерео записываются в разных столбцах массива. Величины должны быть в диапазоне [-1; 1];

**wavwrite(Y, FS, WAVEFILE)** – делает то же с заданием частоты дискретизации FS (в герцах);

### Пример

АМ с подавленной несущей осуществляется функцией `ssbmod` пакета Communications. При этом используется формирование сигнала с помощью преобразования Гильберта, выполняемого в частотной области с использованием функции `hilbert` пакета Signal Processing.

По умолчанию функция `ssbmod` формирует сигнал с нижней боковой полосой. Для формирования сигнала с верхней боковой полосой необходимо указать дополнительный параметр - строку 'upper'. Перед этим параметром следует еще один необязательный параметр - начальная фаза несущего колебания `ini_phase`. Поэтому значение этой фазы при формировании сигнала с верхней боковой полосой необходимо обязательно задать - хотя бы в виде пустой матрицы (для использования значения по умолчанию, равного нулю).

Реализуем однополосную модуляцию для сформированного ранее модулирующего ЛЧМ-сигнала. Создадим однополосные сигналы с нижней и верхней боковой полосой и построим их спектрограммы:

```
Fs = 8e3; % частота дискретизации
T = 2; % длительность сигнала
t = 0:1/Fs:T;% вектор значений времени
f0 = 0;% начальная частота
f1 = 1e3; % конечная частота
Fc = 2e3; % частота несущей
s_M=chirp(t, f0, T, f1); % ЛЧМ-сигнал
s_SSB_LOW = ssbmod(s_M, Fc, Fs);
s_SSB_HI = ssbmod(s_M, Fc, Fs, [], 'upper');
figure
specgram(s_SSB_LOW, [], Fs)
figure
specgram(s_SSB_HI, [], Fs)
```

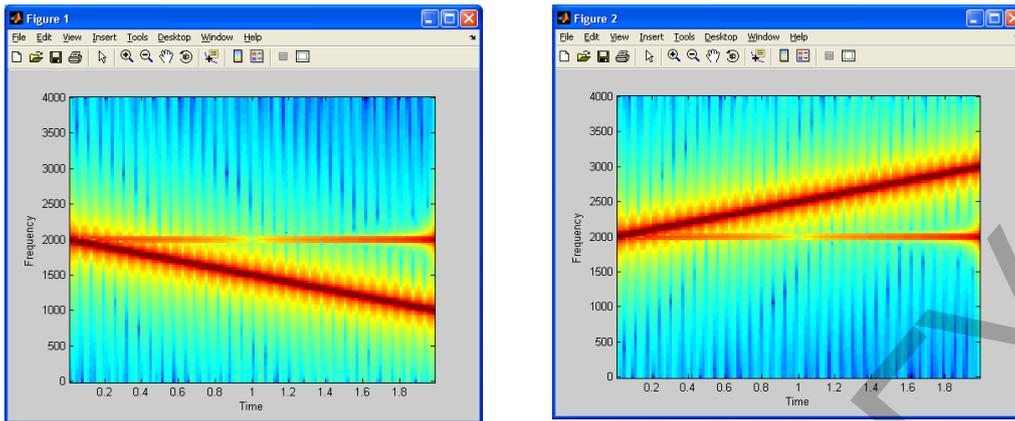


Рисунок 2'. Спектрограммы сигналов

### Задание

1. Выполнить пример.

2. На рисунке 3 приведен простейший пример технологии FDM.

В данном случае реализована схема с тремя каналами передачи речи. В канале 1 голосовой сигнал из диапазона 300-3400 Гц модулируется сигналом с частотой 20 кГц. В каналах 2 и 3 аналогичный голосовой сигнал модулируется сигналами с частотами 16 и 12 кГц. В приведенном примере сохраняются лишь нижние боковые полосы. Результатом смешивания и фильтрации (для удаления верхних боковых полос) являются сдвинутые по частоте сигналы, показанные на рисунке 3. Суммарный выходной сигнал есть суммой трех сигналов и принадлежит диапазону 8,6-19,7 кГц. Выполнить частотное уплотнение сигнала с помощью амплитудной модуляции с одной полосой, построить спектрограммы и график спектральной плотности сигнала. Демодулировать полученный сигнал.

3. Выполнить те же пункты для тонального сигнала с шумом Hello.wav. Записать и прослушать результат.

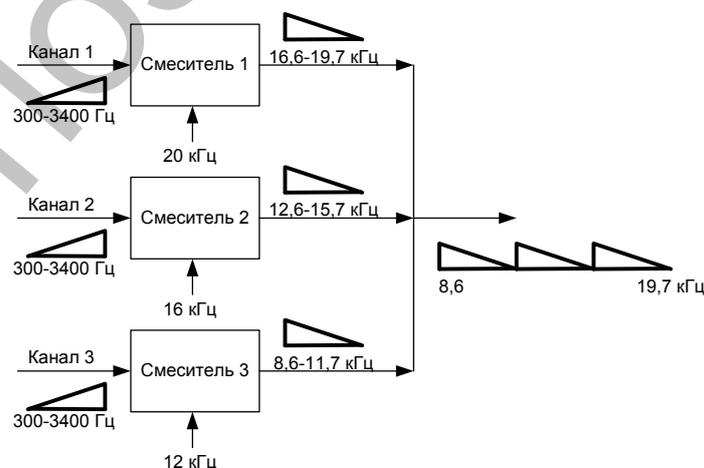


Рисунок 3. Простейший пример FDM.  
Три сдвинутых по частоте канала передачи речи

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Сергиенко А.Б. Цифровая обработка сигналов. – СПб.: Питер, 2002. – 608 с.
2. Скляр Б. Цифровая связь. Теоретические основы и практическое применение. – М.: «Вильямс», 2007. – С. 1104.
3. Солонина А.И., Улахович Д.А. и др. Основы цифровой обработки сигналов: курс лекций. – СПб.: БХВ-Петербург, 2003. – 594 с.
4. Очерк А.Б. Сергиенко «Цифровая модуляция» [Электронный ресурс]. – 2016. – Режим доступа: <http://matlab.exponenta.ru/communication/book4/index.php>. – Дата доступа: 01.12.2016.
5. Block encoder [Electronic resource]. – Mode of access: <https://www.mathworks.com/help/comm/ref/encode.html>. – Date of access: 01.12.2016.

Учебное издание

**КРАСНОБАЕВ** Евгений Алексеевич

**БУЙНОВ** Николай Сергеевич

**СИСТЕМЫ СВЯЗИ И СЕТИ ПЕРЕДАЧИ ИНФОРМАЦИИ**

Методические рекомендации  
к выполнению лабораторных работ

Технический редактор

*Г.В. Разбоева*

Компьютерный дизайн

*Е.А. Барышева*

Подписано в печать . . . . . 2017. Формат 60x84<sup>1</sup>/<sub>16</sub>. Бумага офсетная.

Усл. печ. л. 1,86. Уч.-изд. л. 1,45. Тираж экз. Заказ . . . . .

Издатель и полиграфическое исполнение – учреждение образования  
«Витебский государственный университет имени П.М. Машерова».

Свидетельство о государственной регистрации в качестве издателя,  
изготовителя, распространителя печатных изданий

№ 1/255 от 31.03.2014 г.

Отпечатано на ризографе учреждения образования  
«Витебский государственный университет имени П.М. Машерова».

210038, г. Витебск, Московский проспект, 33.