

Отчет по лабораторным работам по  
дисциплине ТСС

Климов С.А., Назарова К.Е.

2014

## Содержание

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>Система верстки <math>\text{\TeX}</math>и расширения <math>\text{\LaTeX}</math></b> | <b>3</b>  |
| 1.1      | Цель работы . . . . .  | 3         |
| 1.2      | Ход работы . . . . .   | 3         |
| 1.3      | Выводы . . . . .   | 3         |
| 1.4      | Материалы . . . . .  | 3         |
| 1.5      | Инструменты . . . . .  | 3         |
| <b>2</b> | <b>Визуализация сигналов во временной и частотной области</b>                          | <b>4</b>  |
| 2.1      | Цель работы . . . . .  | 4         |
| 2.2      | Постановка задачи . . . . .  | 4         |
| 2.3      | Справочные материалы . . . . .   | 4         |
| 2.4      | Ход работы . . . . .   | 4         |
| 2.5      | Результаты работы . . . . .  | 5         |
| 2.6      | Выводы . . . . .   | 7         |
| <b>3</b> | <b>Спектры простых сигналов</b>  | <b>9</b>  |
| 3.1      | Цель работы . . . . .  | 9         |
| 3.2      | Постановка задачи . . . . .  | 9         |
| 3.3      | Справочные материалы . . . . .   | 9         |
| 3.4      | Ход работы . . . . .   | 9         |
| 3.5      | Результаты работы . . . . .  | 10        |
| 3.6      | Выводы . . . . .   | 15        |
| <b>4</b> | <b>Линейная фильтрация</b>   | <b>16</b> |
| 4.1      | Цель работы . . . . .  | 16        |
| 4.2      | Постановка задачи . . . . .  | 16        |
| 4.3      | Справочные материалы . . . . .   | 16        |
| 4.4      | Ход работы . . . . .   | 16        |
| 4.5      | Результаты работы . . . . .  | 17        |
| 4.6      | Выводы . . . . .   | 20        |
| <b>5</b> | <b>Аналоговая модуляция</b>  | <b>21</b> |
| 5.1      | Цель работы . . . . .  | 21        |
| 5.2      | Постановка задачи . . . . .  | 21        |
| 5.3      | Справочные материалы . . . . .   | 21        |
| 5.4      | Ход работы . . . . .   | 21        |
| 5.5      | Выводы . . . . .   | 29        |
| <b>6</b> | <b>Частотная и фазовая модуляция</b>   | <b>30</b> |
| 6.1      | Цель работы . . . . .  | 30        |
| 6.2      | Постановка задачи . . . . .  | 30        |
| 6.3      | Справочные материалы . . . . .   | 30        |
| 6.4      | Ход работы . . . . .   | 30        |
| 6.5      | Выводы . . . . .   | 35        |
| 6.6      | Выводы . . . . .   | 38        |

# 1 Система верстки $\text{\TeX}$ и расширения $\text{\LaTeX}$

## 1.1 Цель работы

Изучение принципов верстки  $\text{\TeX}$ , создание первого отчета

## 1.2 Ход работы

### Изучение

1. Создание минимального файла .tex в простом текстовом редакторе - преамбула, тело документа
2. Компиляция в командной строке - latex, xdv, pdflatex
3. Оболочка TexMaker, Быстрый старт, быстрая сборка
4. Создание титульного листа, нескольких разделов, списка, несложной формулы

### Выполнение практического задания

$$c^2 = a^2 + b^2$$

$$c_1 + c_2 = b_0 \frac{2 * a}{\log 2} \sum_{i=0}^{\infty} \Theta \quad (1)$$

$$b(k) = (-1)^k \frac{2}{N} \sum_{n=0}^{\frac{N}{2}-1} a(n) \cos \frac{\pi k(2n+1)}{N} \quad (2)$$

$$R_2 = \frac{(|A| + 1)/(\pi f_c C_1)}{b + \sqrt{b^2 - 4c(|A| + 1)C_2/C_1}} \quad (3)$$

## 1.3 Выводы

При выполнении данной работы мы ознакомились с ситемой верстки  $\text{\TeX}$ и расширением  $\text{\LaTeX}$ . Были получены начальные навыки верстки документа, а также построения различного вида формул.

## 1.4 Материалы

1. Не очень краткое введение
2. Конспект-справочник
3. <http://www.inp.nsk.su/~baldin/LaTeX/lurs.pdf>
4. Математика в  $\text{\LaTeX}$

## 1.5 Инструменты

1.  $\text{\TeX}$  для Windows ProText
2. TeXnicCenter
3. Texmaker

## 2 Визуализация сигналов во временной и частотной области

### 2.1 Цель работы

Познакомиться со средствами генерации сигналов и визуализации их спектров.

### 2.2 Постановка задачи

В командном окне MATLAB и в среде Simulink промоделировать чистый синусоидальный сигнал, а также синусоидальный сигнал с шумом. Получить их спектры.

### 2.3 Справочные материалы

В.С. Гутников Фильтрация измерительных сигналов пп.1-2, 11-12

### 2.4 Ход работы

В среде MATLAB вначале моделируем чистый синусоидальный сигнал по формуле:

$$A(t) = A_0 \sin(2\pi ft + f_0)$$

Затем моделируем зашумленный сигнал по формуле:

$$A(t) = A_0 \sin(2\pi ft + f_0) + A_1 \text{rand}()$$

Для выделения частот регулярных составляющих используем преобразование Фурье, которое реализовано встроенной в MATLAB функцией.

#### Код MATLAB

```
f = 10;
x = 0:0.01:8*pi;
y = sin(2*pi*f*x+pi/2);
plot(x(1:200),y(1:200))
grid
figure
spectrum = fft(y,512);
norm_spectrum = spectrum.*conj(spectrum)/512;
f=100*(0:511)/512;
plot(f, norm_spectrum(1:512))
axis([0 max(f) 0 10])
grid
y_noise = y + 0.6 * rand(size(x));
figure
plot(x(1:200),y_noise(1:200));
grid
spectrum_noise = fft(y_noise,512);
noise_spectrum = spectrum_noise.*conj(spectrum_noise)/512;
```

```
figure
plot(f, noise_spectrum(1:512))
axis([0 max(f) 0 10])
grid
```

## 2.5 Результаты работы

В результате получаем следующие графики временных и частотных характеристик для чистого и зашумленного синусоидального сигналов:

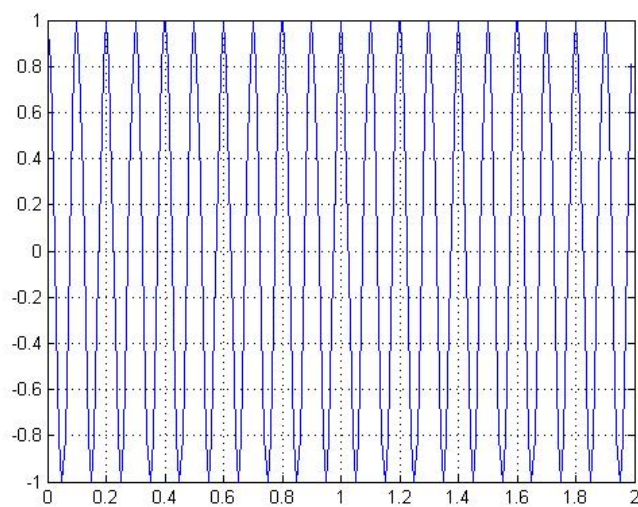


Рис. 1: Временная характеристика чистого синусоидального сигнала

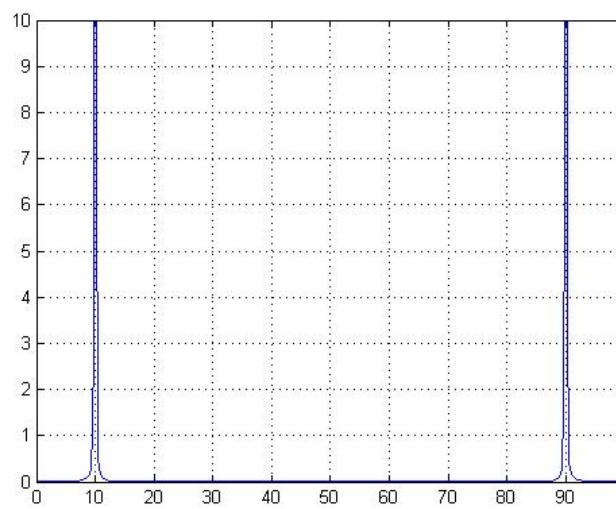


Рис. 2: Частотная характеристика чистого синусоидального сигнала

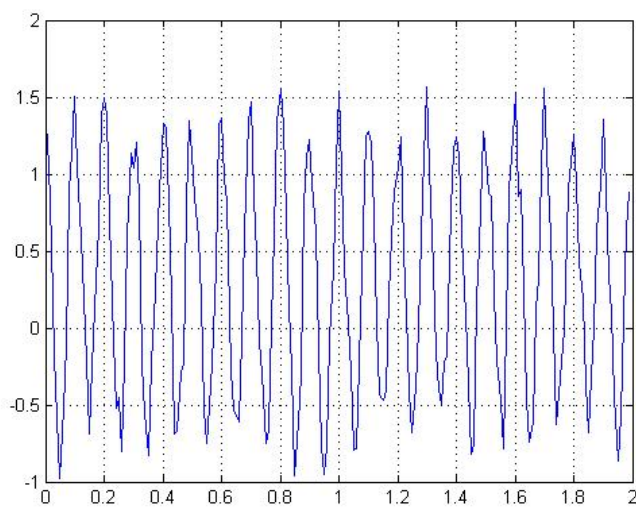


Рис. 3: Временная характеристика зашумленного синусоидального сигнала

## 2.6 Выводы

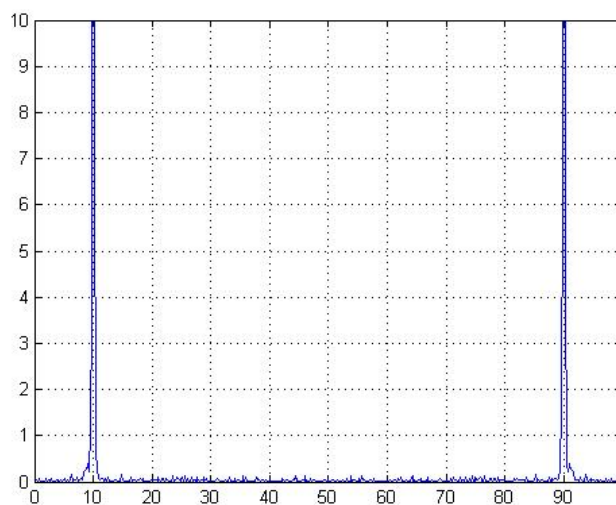


Рис. 4: Частотная характеристика зашумленного синусоидального сигнала

По результатам моделирования синусоидальных сигналов (чистого и с шумом) а также их полученных спектров можно сделать следующие выводы: умноженный на свое комплексное сопряженное спектр является нормированным, синус не бесконечен, спектр испытывает свертку с синком, повторение спектра происходит на частоте, кратной частоте дискретизации.



## 3 Спектры простых сигналов

### 3.1 Цель работы

Получить представление о свойствах спектров.

### 3.2 Постановка задачи

В командном окне MATLAB и в среде Simulink промоделировать следующие тестовые сигналы:

- Полигармонический сигнал

$$y(t) = \sum_{n=0}^{N-1} \cos(nt) \quad (4)$$

- Прямоугольный импульсный сигнал

$$y(t) = \prod(t, T_i) \quad (5)$$

- Треугольный импульсный сигнал

$$y(t) = \Delta(t, T_i) \quad (6)$$

и получить их спектры.

### 3.3 Справочные материалы

В.С. Гутников. Фильтрация измерительных сигналов ш.3-6, 13-14

### 3.4 Ход работы

Для моделирования заданных сигналов использовался приведенный код MATLAB:

```
t = 0:0.01:4*pi;
N = 100;
y = 0;
% Полигармонический сигнал
y = sin(pi*t)+sin(2*pi*t)+sin(pi*0.3*t);
plot(t,y,'LineWidth',2)
grid
figure
spectrum = fft(y,512);
norm_spectrum = spectrum.*conj(spectrum)/512;
f=100*(0:255)/512;
plot(f, norm_spectrum(1:256),'LineWidth',2)
axis([0 max(f) 0 10])
grid
% Прямоугольный сигнал
figure
```

```

y1 = square(t,50);
plot(t(1:100),y1(1:100),'LineWidth',2);
ylim([-2,2]);
grid
figure
spectrum = fft(y1,512);
norm_spectrum = spectrum.*conj(spectrum)/512;
f1=100*(0:255)/512;
plot(f1, norm_spectrum(1:256),'LineWidth',2)
axis([0 max(f1) 0 10])
grid
% Треугольный сигнал
figure
y2 = conv(square(t,20),square(t,20));
plot(t(1:100),y2(1:100),'LineWidth',2);
grid
figure
spectrum = fft(y2,512);
norm_spectrum = spectrum.*conj(spectrum)/512;
f2=100*(0:255)/512;
plot(f2, norm_spectrum(1:256)/1000,'LineWidth',2)
axis([0 max(f2) 0 50])
grid

```

### 3.5 Результаты работы

В результате были получены следующие сигналы и их спектры:

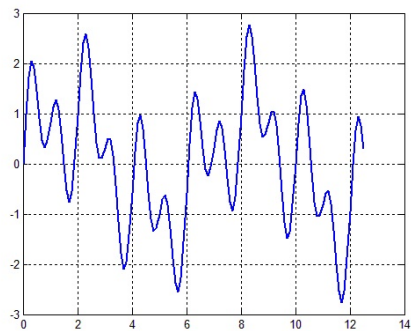


Рис. 5: Полигармонический сигнал

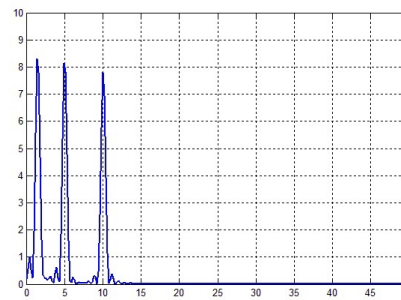


Рис. 6: Спектр полигармонического сигнала

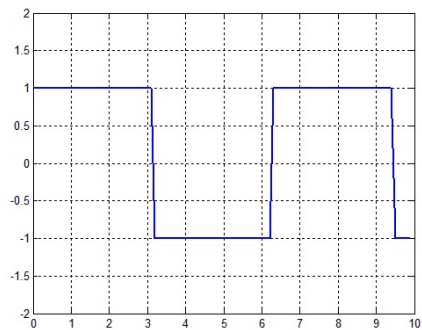


Рис. 7: Полигармонический сигнал

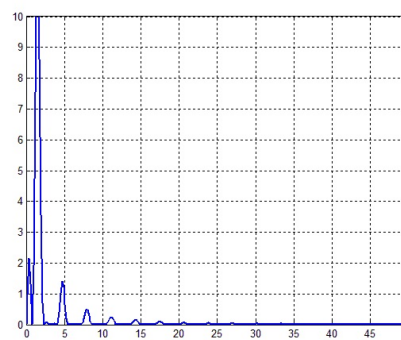


Рис. 8: Спектр полигармонического сигнала

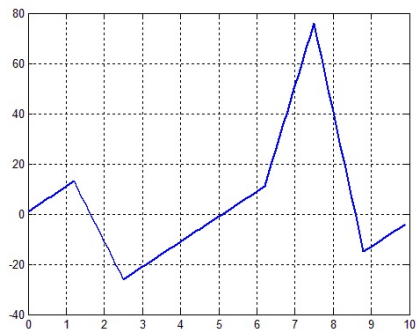


Рис. 9: Полигармонический сигнал

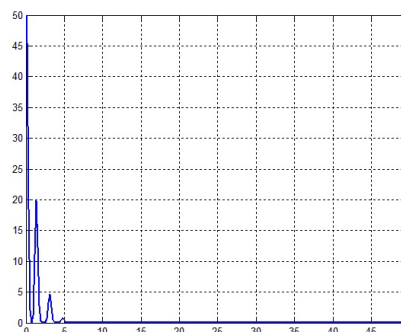


Рис. 10: Спектр полигармонического сигнала

В ходе моделирования сигналов в Simulink были собраны следующие схемы и получены соответствующие графики:

- Полигармонический сигнал (Рис. 11 - 13)
- Прямоугольный импульсный сигнал (Рис. 14 - 16)
- Треугольный импульсный сигнал (Рис. 17 - 19)

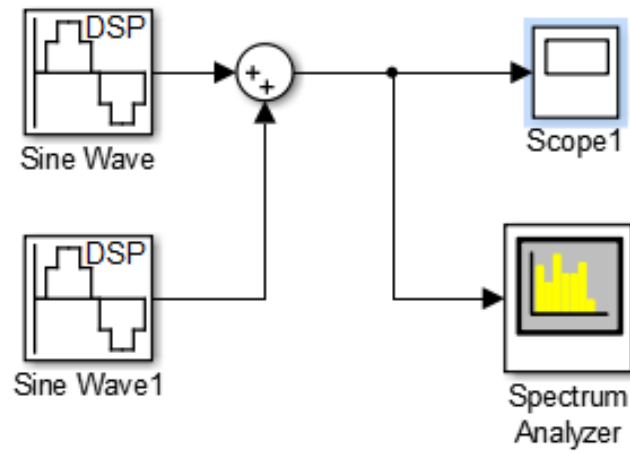


Рис. 11: Полигармонический сигнал (Simulink)

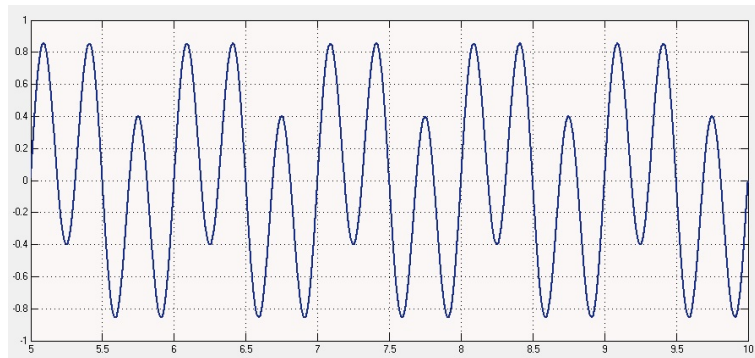


Рис. 12: Полигармонический сигнал

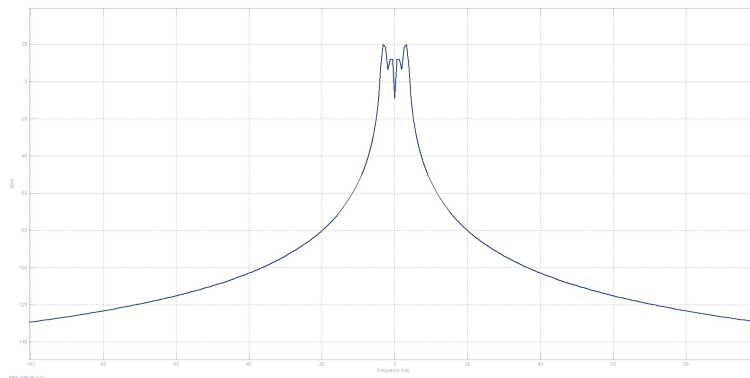


Рис. 13: Спектр полигармонического сигнала

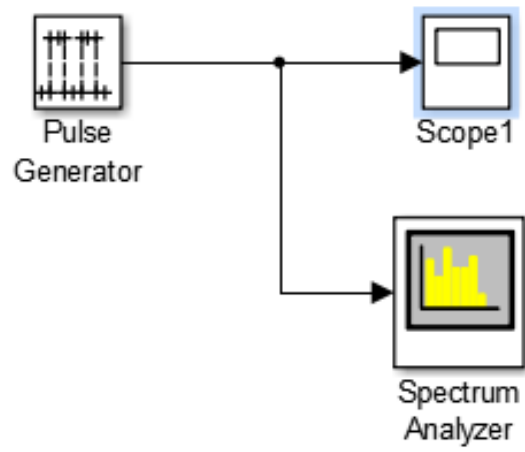


Рис. 14: Прямоугольный импульсный сигнал (Simulink)

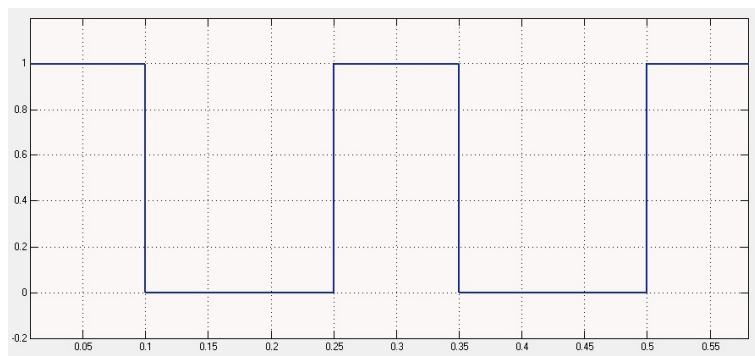


Рис. 15: Прямоугольный импульсный сигнал

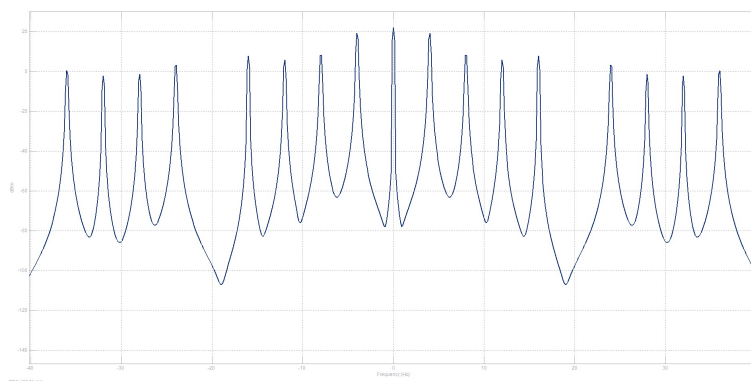


Рис. 16: Спектр прямоугольного сигнала

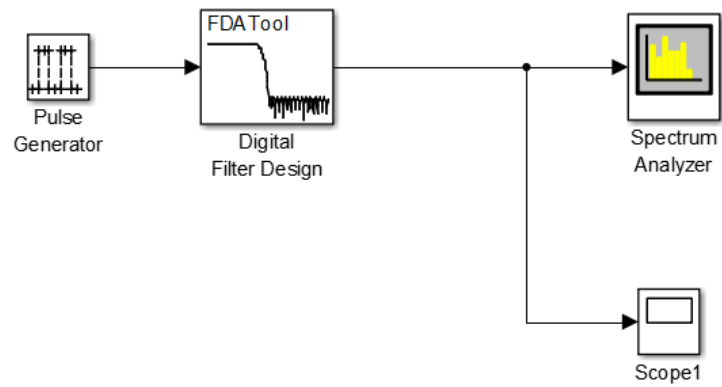


Рис. 17: Треугольный импульсный сигнал (Simulink)

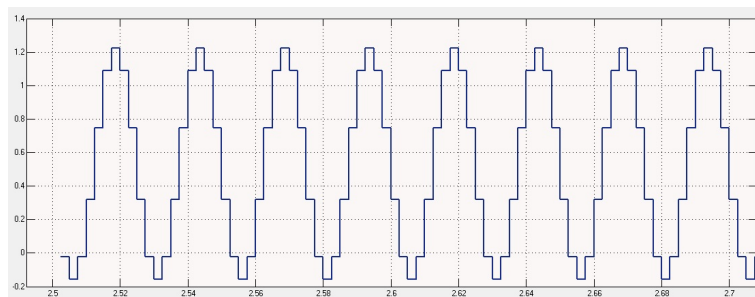


Рис. 18: Треугольный импульсный сигнал

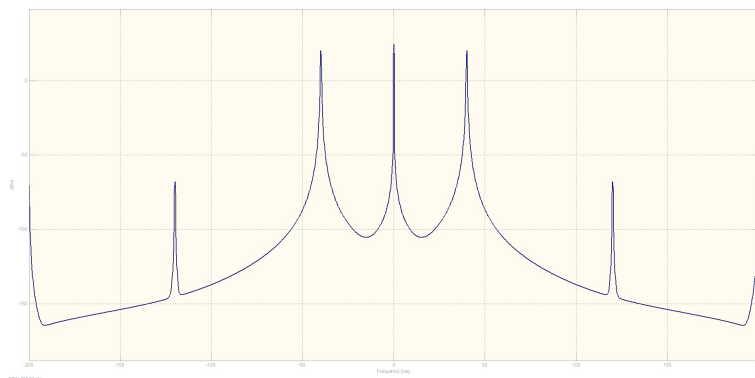


Рис. 19: Спектр треугольного сигнала

### 3.6 Выводы

В результате проделанной работы было проведено моделирование в среде MatLAB и Simulink полигармонического, прямоугольного и треугольного импульсных сигналаов, а также получены их спектры. В Simulink для получения генератора треугольного сигнала используется генератор прямоугольных импульсов каскадно с фильтром с прямоугольным окном. Это обоснуется тем, что свертка двух прямоугольных импульсов в результате дает треугольный.

## 4 Линейная фильтрация

### 4.1 Цель работы

Изучить воздействие ФНЧ на тестовый сигнал с шумом.

### 4.2 Постановка задачи

Сгенерировать гармонический сигнал с шумом и синтезировать ФНЧ. Получить сигнал во временной и частотной областях до и после фильтрации. Сделать выводы о воздействии ФНЧ на спектр сигнала.

### 4.3 Справочные материалы

В.С. Гутников. Фильтрация измерительных сигналов шп. 17-19

### 4.4 Ход работы

Фильтрация сигнала - изменение его спектра, которое обычно применяется с целью увеличить отношение полезного сигнала к шумам и помехам или усилить какие-нибудь полезные качества сигнала. В данной работе рассматривается фильтр нижних частот (ФНЧ), пропускающий низкочастотные составляющие спектра и задерживающий высокочастотные, а именно фильтр Баттерворта с максимально гладкой АЧХ. Для моделирования заданных сигналов использовался приведенный ниже код MATLAB:

```
x = 0:0.01:4*pi;
f = 100*(0:255)/512;
noise = rand(size(x));
y = sin(2*pi*x); % Сигнал без шума
y_noisy = y+0.3*noise; % Сигнал с шумом
figure
plot(x(1:200),y(1:200))
grid
figure
plot(x(1:200),y_noisy(1:200))
grid
[B,A] = butter(16,0.98); % Синтез ФНЧ Баттерворта
B = B./sum(B);
A = A./sum(A);
y_filtered = conv(y_noisy, [B, A]); % Обработка сигнала ФНЧ
figure
plot(x(1:200),y_filtered(1:200))
grid
noisy_spectrum = fft(y_noisy, 512); % Спектр сигнала с шумом
norm_noisy_spectrum = noisy_spectrum.*conj(noisy_spectrum)/512;
figure
plot(f, norm_noisy_spectrum(1:256))
axis([0 max(f) 0 2])
grid
filtered_spectrum = fft(y_filtered, 512); % Спектр фильтрованного сигнала
```



```

norm_filtered_spectrum = filtered_spectrum.*conj(filtered_spectrum)/512;
figure
plot(f, norm_filtered_spectrum(1:256))
axis([0 max(f) 0 2])
grid

```

## 4.5 Результаты работы

В результате работы представленного кода были получены следующие данные:

- Исходный гармонический сигнал (Рис. 20);
- Сигнал с шумом и его спектр (Рис. 21 - 22);
- Отфильтрованный сигнал и его спектр (Рис. 23 - 24).

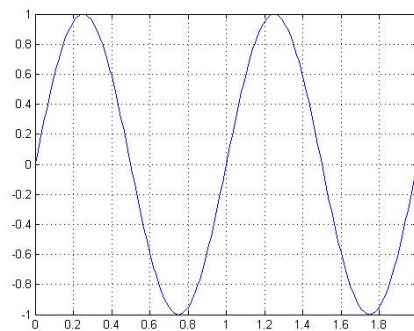


Рис. 20: Исходный гармонический сигнал

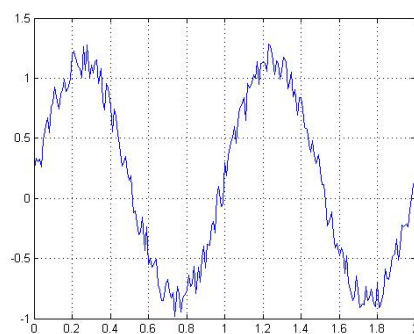


Рис. 21: Зашумленный гармонический сигнал

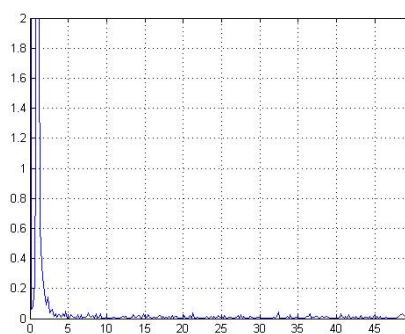


Рис. 22: Спектр зашумленного сигнала

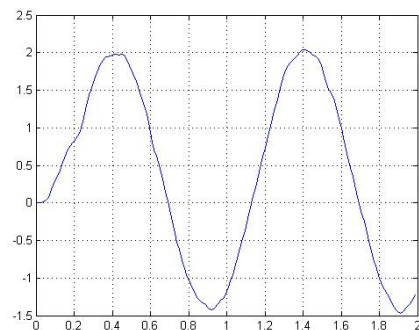


Рис. 23: Отфильтрованный сигнал

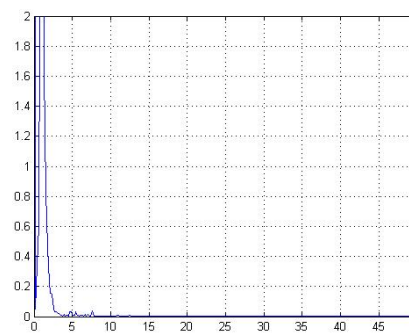


Рис. 24: Спектр отфильтрованного сигнала

Аналогичное моделирование было проведено в среде Simulink (построенный блок отображен на Рис. 25). В результате были получены характеристики, представленные на Рис. 26 - 27.

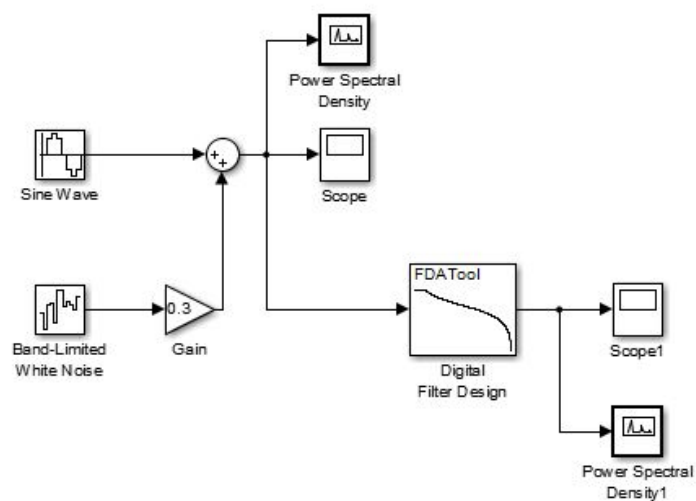


Рис. 25: Исходный гармонический сигнал

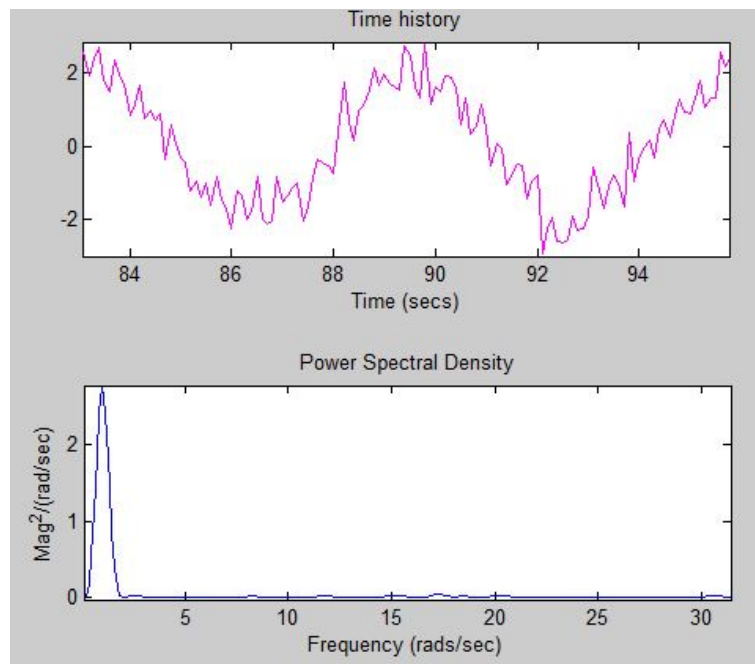


Рис. 26: Отфильтрованный сигнал

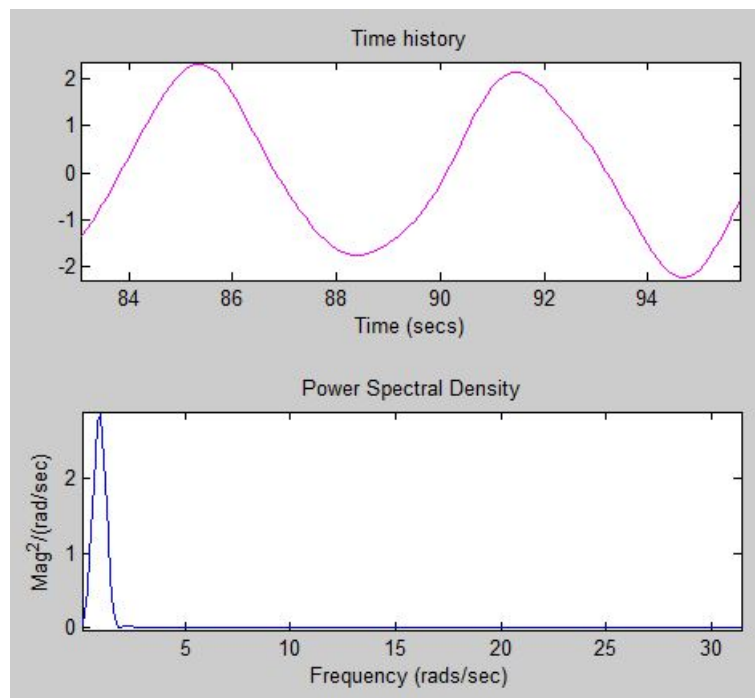


Рис. 27: Спектр отфильтрованного сигнала

## 4.6 Выводы

В данной работе было изучено воздействие ФНЧ на тестовый сигнал с шумом, а именно произведено зашумление и удаление шума линейным фильтром Баттерворта. Однако фильтрация оказалась неполной. Это связано с тем, что использованный линейный фильтр не способен разделить шум и полезный сигнал в одной частотной области. Для полной фильтрации необходимо использовать идеальный фильтр с прямоугольным окном.

## 5 Аналоговая модуляция

### 5.1 Цель работы

Изучение аналоговой модуляции/демодуляции сигнала.

### 5.2 Постановка задачи

1. Сгенерировать однотоальный сигнал низкой частоты.
2. Выполнить амплитудную модуляцию сигнала по закону

$$u(t) = (1 + MU_m \cos(\Omega t)) \cos(\omega_0 t + \phi_0). \quad (7)$$

3. Получить спектр модулированного сигнала.
4. Выполнить модуляцию с подавлением несущей

$$u(t) = MU_m \cos(\omega t) \cos(\omega_0 t + \phi_0). \quad (8)$$

Получить спектр.

5. Выполнить однополосную модуляцию:

$$u(t) = U_m \cos(\omega t) \cos(\omega_0 t + \phi_0) + \frac{U_m}{2} \sum_{n=1}^N M_n (\cos(\omega_0 + \Omega_n)t + \phi_0 + \Phi_n), \quad (9)$$

положив  $n = 1$ .

6. Выполнить синхронное детектирование и получить исходный однополосный сигнал.
7. Рассчитать КПД модуляции

$$\eta_A M = \frac{U_m^2 M^2 / 4}{P_U} = \frac{M^2}{M^2 + 2}. \quad (10)$$

### 5.3 Справочные материалы

Н.В. Богач и др. Обработка сигналов в информационных системах, с. 110-118, 125-127.

### 5.4 Ход работы

АМ - сигнал представляет собой произведение информационной огибающей  $u(t)$  и гармонического колебания ее заполнения с более высокими частотами. Простейшая форма модулированного сигнала создается при *однотоальной* АМ - модуляции несущего сигнала гармоническим колебанием с одной частотой  $\Omega$ :

$$u(t) = (1 + MU_m \cos(\Omega t)) \cos(\omega_0 t + \phi_0). \quad (11)$$

Реализация аналоговой модуляции с помощью MATLAB:

```

x = 0:0.01:4*pi;
f0 = 0.2;
% Исходный сигнал
y = sin(2*pi*f0*x);
figure
plot(x(1:500),y(1:500), 'LineWidth', 2)
grid
% Спектр исходного сигнала
spectrum = fft(y, 512);
norm_spectrum = spectrum.*conj(spectrum)/512;
f = 100*(0:255)/512;
figure
plot(f, norm_spectrum(1:256), 'LineWidth', 2)
axis([0 max(f) 0 40])
grid
% Амплитудная модуляция
Fc = 5*f0;
Fs = 50*f0;
U = ammod(y, Fc, Fs, 0, 1);
figure
plot(x(1:500), U(1:500))
grid
% Спектр модулированного сигнала
u_spectrum = fft(U, 512);
norm_u_spectrum = u_spectrum.*conj(u_spectrum)/512;
figure
plot(f, norm_u_spectrum(1:256))
axis([0 max(f) 0 30])
grid

```

В результате выполнения приведенного кода были получены следующие характеристики:

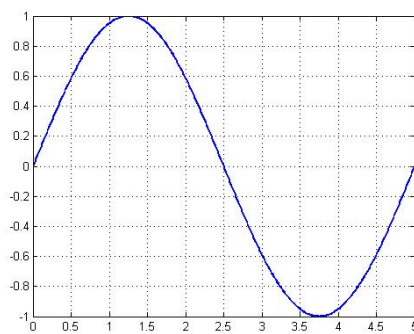


Рис. 28: Исходный гармонический сигнал

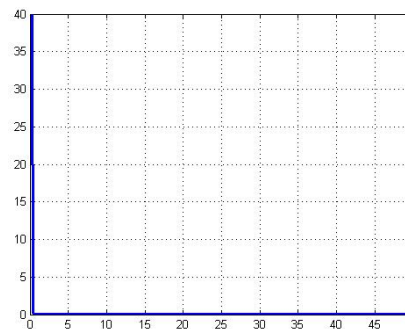


Рис. 29: Спектр исходного сигнала

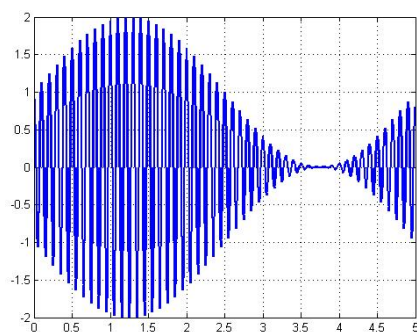


Рис. 30: Модуляция с  $M = 1$

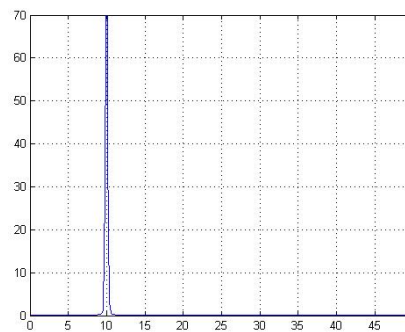


Рис. 31: Спектр АМ-сигнала ( $M = 1$ )

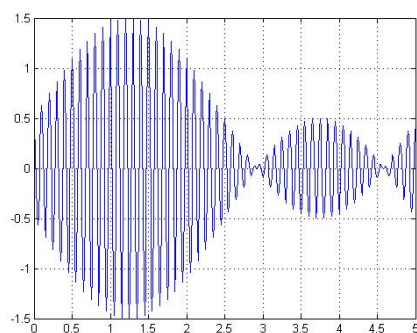


Рис. 32: Модуляция с  $M = 2$

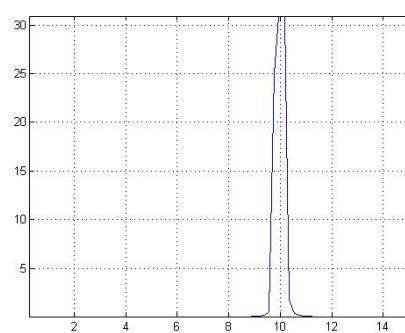


Рис. 33: Спектр АМ-сигнала ( $M = 2$ )

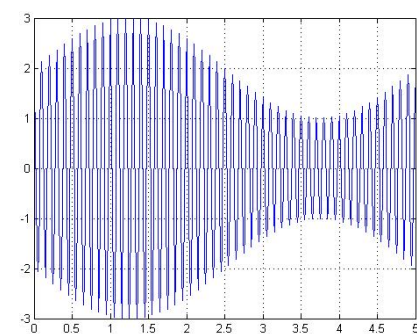


Рис. 34: Модуляция с  $M = 0.5$

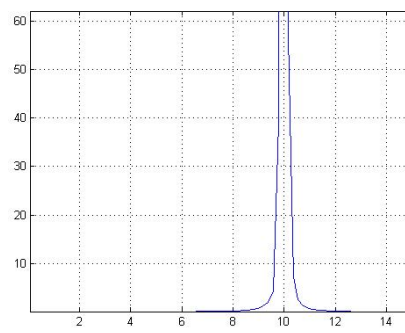


Рис. 35: Спектр АМ-сигнала ( $M = 0.5$ )

При балансной модуляции (АМ с подавлением несущей) производится перемножение двух сигналов - модулирующего и несущего, при котором происходит подавление несущего колебания, соответственно, КПД модуляции становится равным 100%. Физическая сущность подавления несущей заключается в том, что при переходе огибающей биений  $U(t)$  через 0 фаза несущей частоты высокочастотного заполнения изменяется на  $180^\circ$ .

Реализация аналоговой модуляции с подавлением несущей с помощью MATLAB:

```
% Амплитудная модуляция с подавлением несущей
Fc = 10*f0;
Fs = 100*f0;
U = ammod(y, Fc, Fs);
figure
plot(x(1:500), U(1:500), 'LineWidth', 2)
grid
% Спектр модулированного сигнала
u_spectrum = fft(U, 512); norm_u_spectrum =
u_spectrum.*conj(u_spectrum)/512;
figure
plot(f, norm_u_spectrum(1:256))
axis([0 max(f) 0 30])
grid
```

В результате выполнения приведенного кода были получены следующие характеристики:

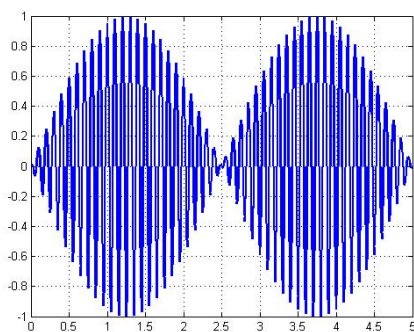


Рис. 36: АМ-сигнал с подавлением несущей

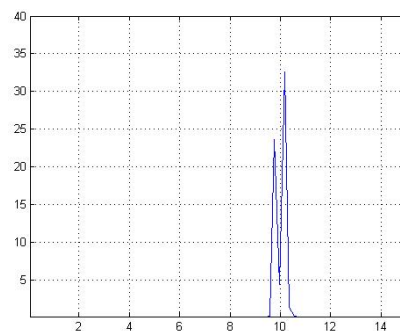


Рис. 37: Спектр АМ-сигнала с подавлением

Спектры двух боковых полос АМ-сигнала являются зеркальным отражением друг друга, т.е. они несут одну и ту же информацию. Поэтому одну из боковых полос можно удалить.

Для этого был использован следующий код MATLAB:

```
% Однополосная амплитудная модуляция
Fc = 10*f0;
Fs = 100*f0;
```



```

U = ssbmod(y, Fc, Fs, [], 'upper');
figure
plot(x(1:500), U(1:500), 'LineWidth', 2)
grid
% Спектр модулированного сигнала
u_spectrum = fft(U, 512); norm_u_spectrum =
u_spectrum.*conj(u_spectrum)/512;
figure
plot(f, norm_u_spectrum(1:256))
axis([0 max(f) 0 30])
grid

```

В результате выполнения приведенного кода были получены следующие графики:

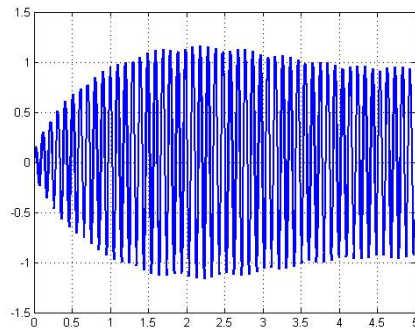


Рис. 38: Однополосный АМ-сигнал

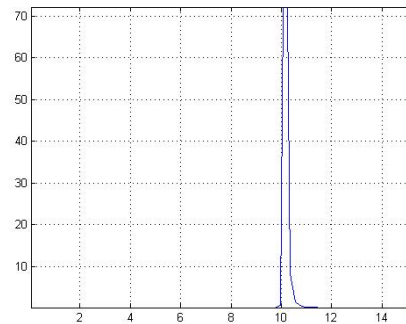


Рис. 39: Спектр однополосного АМ-сигнала

Синхронное детектирование является одним из способов демодуляции АМ-сигнала. Его суть состоит в умножении частоты сигнала на опорное колебание с несущей частотой. Результат умножения содержит два слагаемых: искомая амплитуда и АМ-сигнал с несущей частотой  $2\omega_0$ , который легко удаляется путем пропускания сигнала через ФНЧ.

Ниже приведен реализующий синхронное детектирование код MATLAB:

```

% Синхронное детектирование
z = ssbdemod(U, Fc, Fs, 0, b, a);
figure
plot(x(1:1000), z(1:1000))
grid
% Спектр демодулированного сигнала
du_spectrum = fft(U, 512);
norm_du_spectrum = du_spectrum.*conj(du_spectrum)/512;
figure
plot(f, norm_du_spectrum(1:256))
axis([0 max(f) 0 70])
grid

```

В результате синхронного детектирования демодулированный сигнал полностью совпал с исходным сигналом:

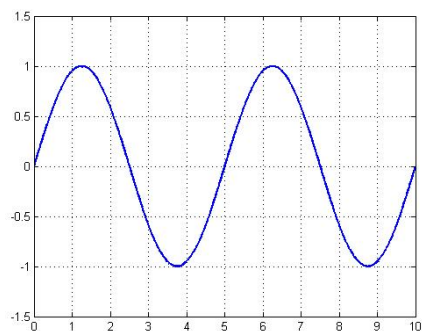


Рис. 40: Демодулированный AM- сигнал

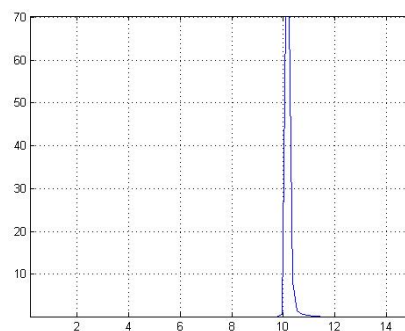


Рис. 41: Спектр демодулированного сигнала

В ходе работы был рассчитан КПД модуляции при различных значениях глубины  $M$ :

- $M = 1$ : 0.33;
- $M = 2$ : 0.67;
- $M = 0.5$ : 0.11.

#### Однополосная модуляция/демодуляция в среде Simulink

Для моделирования амплитудной однополосной модуляции и демодуляции использовались блоки SSB AM Modulator/Passband и SSB AM Demodulator/Passband. Результаты моделирования приведены на Рис. 42-45:

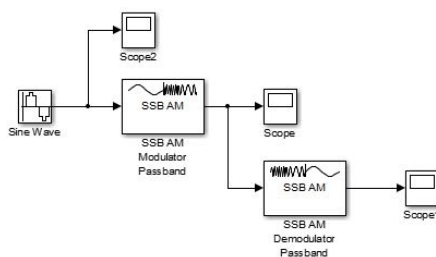


Рис. 42: Схема однополосной модуляции/демодуляции

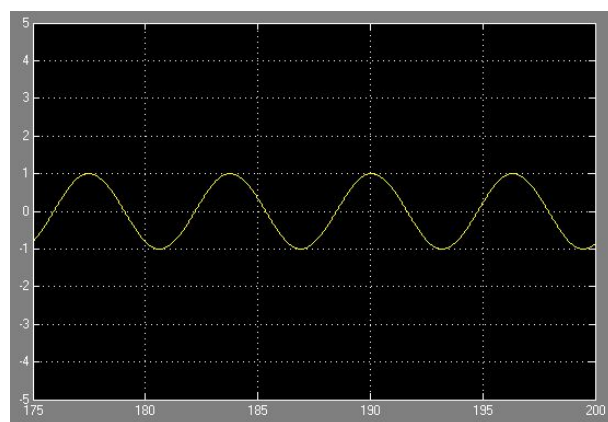


Рис. 43: Исходный сигнал

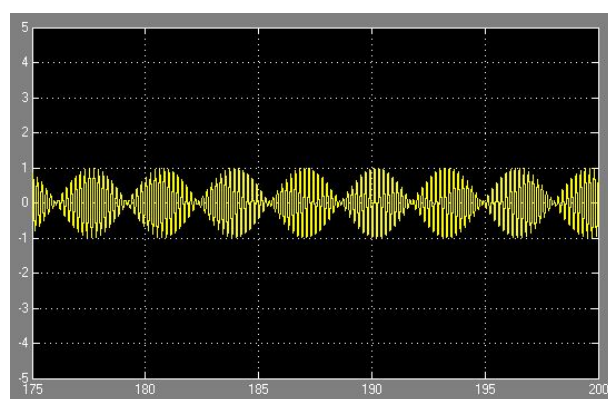


Рис. 44: АМ-сигнал

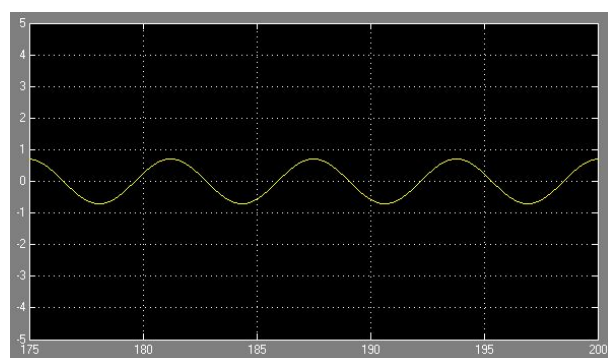


Рис. 45: Демодулированный АМ-сигнал

Для моделирования балансной однополосной модуляции и демодуляции использовались блоки DSBSC AM Modulator/Demodulator Passband. Исходный сигнал тот же (Рис. 43). Результаты моделирования приведены на Рис. 46-48:

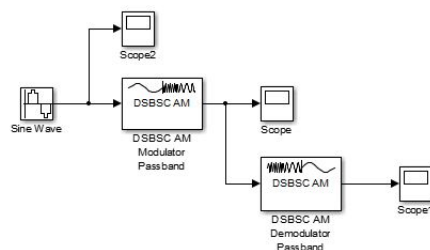


Рис. 46: Схема балансной модуляции/демодуляции

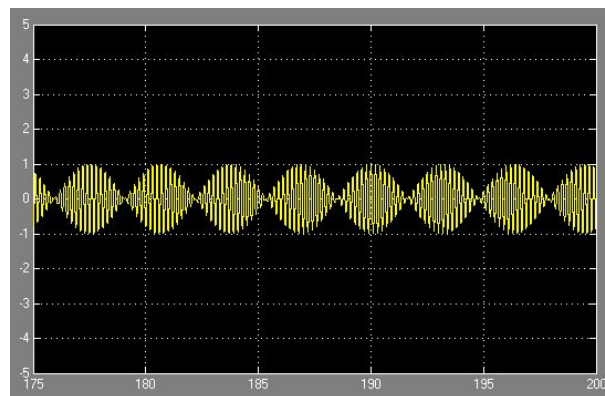


Рис. 47: БАП-сигнал

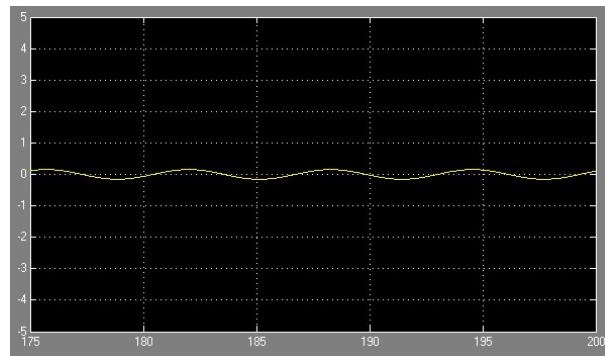


Рис. 48: Демодулированный БАП-сигнал

## 5.5 Выводы

В ходе выполнения лабораторной работы был сгенерирован однотоальный сигнал низкой частоты, выполнена АМ сигнала и получен спектр модулированного сигнала. Также была выполнена БАП и однополосная АМ и получены их спектры. Произведено синхронное детектирование и получен исходный однополосный сигнал. Рассчитан КПД модуляции.

АМ применяется на сравнительно низких частотах (не выше коротких волн). Это обусловлено низким КПД использования энергии модулированных сигналов.

Ширина спектра АМ-сигнала с подавленной несущей, как в случае с обычной АМ, в два раза больше, чем у модулирующего сигнала. Но при БАП производится перемножение двух сигналов – модулирующего и несущего, при котором происходит подавление несущего колебания, соответственно, КПД модуляции становится равным 100%.

Двухполосная АМ с подавленной несущей имеет преимущества перед обычной АМ только в энергетическом плане - за счет устранения несущего колебания, ширина спектра при этом по-прежнему вдвое больше, чем у модулирующего сигнала. Однако спектры двух боковых полос АМ-сигнала являются зеркальным отражением друг друга, поэтому одну из боковых полос можно удалить.

Однополосный сигнал можно представить как сумму двух АМ-сигналов, несущие колебания которых имеют одну и ту же частоту, но сдвинуты по фазе относительно друг друга на  $90^\circ$ .

Синхронное детектирование является одним из способов демодуляции АМ-сигнала. Его суть состоит в умножении частоты сигнала на опорное колебание с несущей частотой. Результат умножения содержит два слагаемых: искомая амплитуда и АМ-сигнал с несущей частотой  $2\omega_0$ , который легко удаляется путем пропускания сигнала через ФНЧ. В нашем случае использовался фильтр Баттерворта.

## 6 Частотная и фазовая модуляция

### 6.1 Цель работы

Изучение частотной и фазовой модуляции/демодуляции сигнала.

### 6.2 Постановка задачи

1. Сгенерировать однотоальный сигнал низкой частоты.
2. Выполнить фазовую модуляцию/демодуляцию сигнала по закону

$$u(t) = U_m \cos(\Omega t + k s(t)), \quad (12)$$

используя встроенную функцию MatLab `pmmmod`, `pmdemod`.

3. Получить спектр модулированного сигнала.
4. Выполнить частотную модуляцию/демодуляцию по закону

$$u(t) = U_m \cos(\omega_0 t + k \int_0^t s(t) dt + \phi_0), \quad (13)$$

используя встроенные функции MatLab `fmmmod`, `fmdemod`.

### 6.3 Справочные материалы

Н.В. Богач и др. Обработка сигналов в информационных системах, с. 118-125, 127-133.

### 6.4 Ход работы

Фазовая модуляция – процесс изменения мгновенной фазы несущего колебания пропорционально изменению непрерывного информационного сигнала. Фазомодулированный сигнал  $s(t)$  имеет следующий вид:

$$s(t) = g(t) \sin[2\pi f_c t + \varphi(t)], \quad (14)$$

где  $g(t)$  — огибающая сигнала;  $\varphi(t)$  является модулирующим сигналом;  $f_c$  — частота несущего сигнала;  $t$  — время. В случае, когда информационный сигнал является дискретным, то говорят о фазовой манипуляции.

По характеристикам фазовая модуляция близка к частотной модуляции. В случае синусоидального модулирующего (информационного) сигнала, результаты частотной и фазовой модуляции совпадают.

Реализация фазовой модуляции/демодуляции с помощью MATLAB:

```
% Исходный сигнал
x = 0:0.01:4*pi;
y = sin(2*pi*x);

figure
subplot(4, 1, 1);
plot(x(1:500), y(1:500))
```

```

grid
title('Исходный сигнал')

% Фазовая модуляция
Fs = 1000; % Частота дискретизации
Fc = 4; % Несущая частота
phasedev = pi/2; % Девияция фазы для фазовой модуляции
pm_y = pmmod(x,Fc,Fs,phasedev); % Фазовая модуляция
subplot(4, 1, 2);
plot(x(1:500), pm_y(1:500))
grid
title('Фазовая модуляция')

spectrum = fft(pm_y, 512); % Спектр фазовой модуляции
norm_spectrum = spectrum.*conj(spectrum)/512;
f = 100*(0:255)/512;
subplot(4, 1, 3);
plot(f, norm_spectrum(1:256))
grid
title('Спектр фазовой модуляции')
axis([min(f) max(f) 0 max(norm_spectrum)]);

pdm_y = pmdemod(pm_y,Fc,Fs,phasedev); % Демодуляция
subplot(4, 1, 4);
plot(x, pdm_y)
grid
title('Фазовая демодуляция')

```

В результате выполнения приведенного кода были получены следующие характеристики:

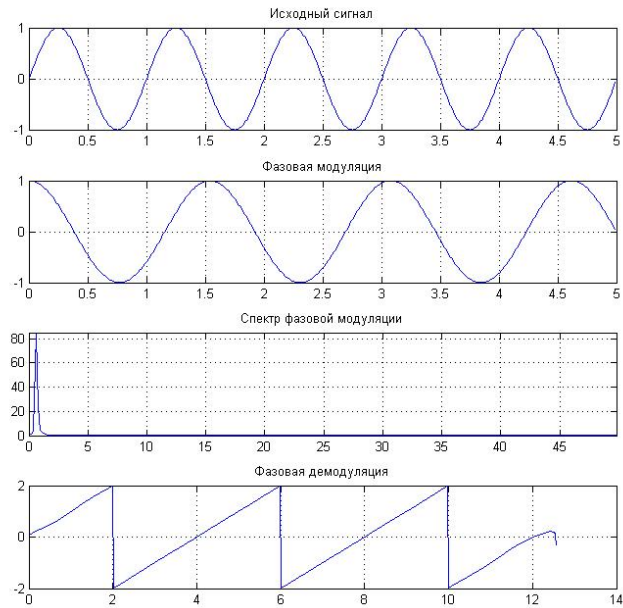


Рис. 49: Результаты фазовой модуляции/демодуляции

Частотная модуляция – процесс изменения мгновенной частоты несущего колебания в соответствии с изменением информационного сигнала. По сравнению с амплитудной модуляцией здесь амплитуда остаётся постоянной.

Реализация частотной модуляции/демодуляции с помощью MATLAB:

```
% Исходный сигнал
x = 0:0.01:4*pi;
y = sin(2*pi*x);

figure
subplot(4, 1, 1);
plot(x(1:500), y(1:500))
grid
title('Исходный сигнал')

% Частотная модуляция
freqdev = 20; % Девияция частоты для частотной модуляции
fm_y = fmod(x,Fc,Fs,freqdev); % Частотная модуляция
subplot(4, 1, 2);
plot(x(1:500), fm_y(1:500))
grid
title('Частотная модуляция')
spectrum = fft(fm_y, 512); % Спектр частотной модуляции
norm_spectrum = spectrum.*conj(spectrum)/512;
```



```

f = 100*(0:255)/512;
subplot(4, 1, 3);
plot(f, norm_spectrum(1:256))
grid
title('Спектр частотной модуляции')
axis([min(f) max(f) 0 max(norm_spectrum)]);
fdm_y = fmdemod(fm_y,Fc,Fs,freqdev); % Демодуляция
subplot(4, 1, 4);
plot(x, fdm_y)
grid
title('Частотная демодуляция')

```

В результате выполнения приведенного кода были получены следующие характеристики:

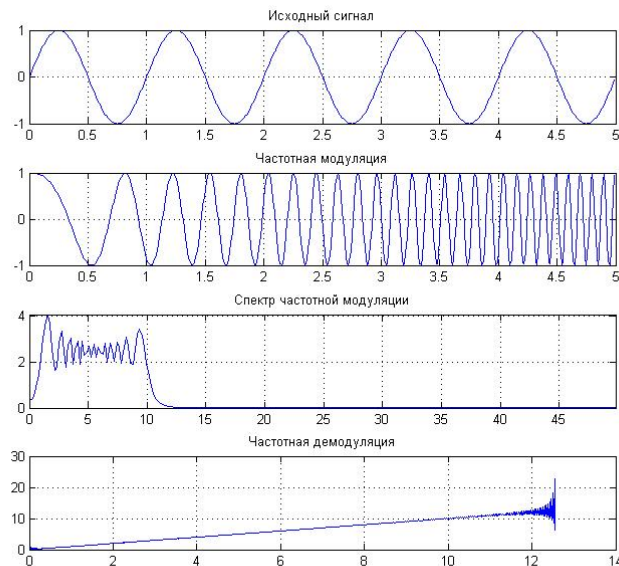


Рис. 50: Результаты частотной модуляции/демодуляции

Для создания модели в Simulink использовался раздел Communication Blockset Simulink. Выполнена модуляция, в том числе с помощью блока захвата фазы Phase-Locked Loop.

Для моделирования использовался исходный сигнал, представленный на Рис. 51.

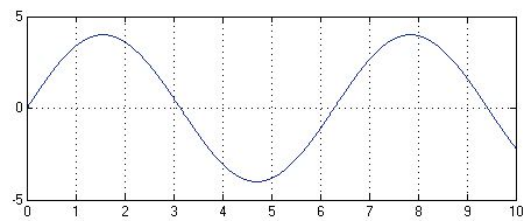


Рис. 51: Исходный сигнал

Фазовая модуляция/демодуляция в Simulink:

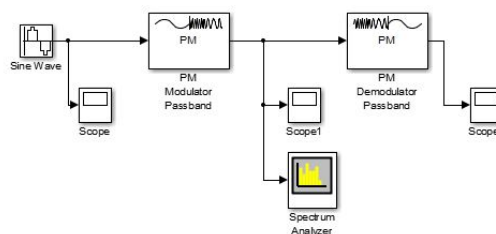


Рис. 52: Фазовая модуляция/демодуляция

В результате были получены следующие характеристики:

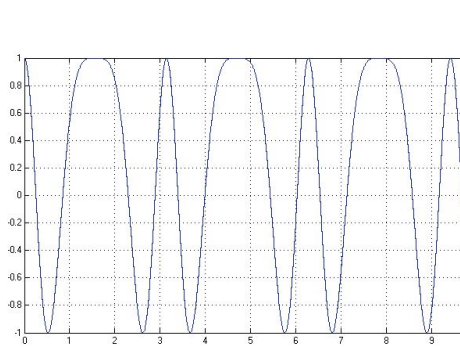


Рис. 53: ФМ-сигнал

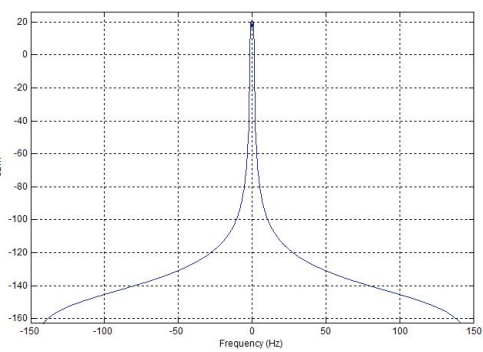


Рис. 54: Спектр ФМ-сигнала

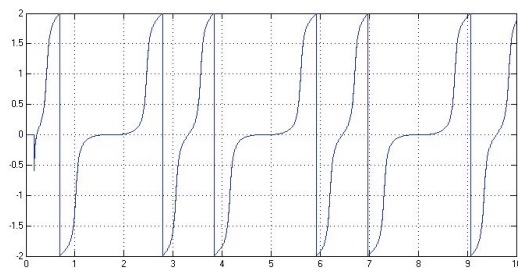


Рис. 55: Думодулированный ФМ-сигнал

## 6.5 Выводы

Частотная модуляция/демодуляция в Simulink:

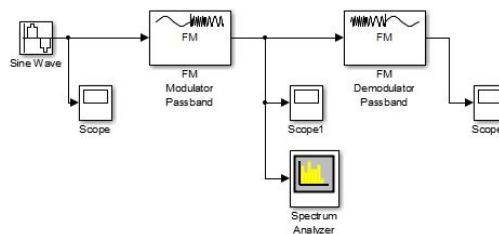


Рис. 56: Частотная модуляция/демодуляция

В результате были получены следующие характеристики:

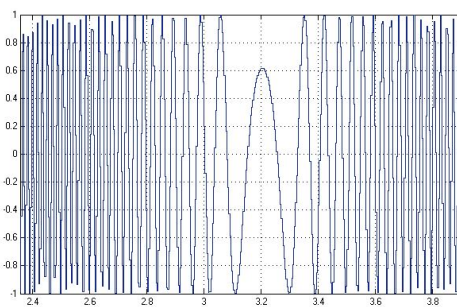


Рис. 57: ЧМ-сигнал

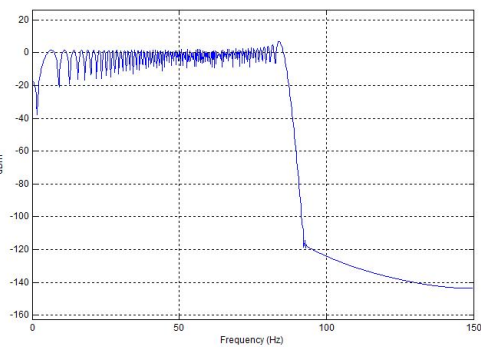


Рис. 58: Спектр ЧМ-сигнала

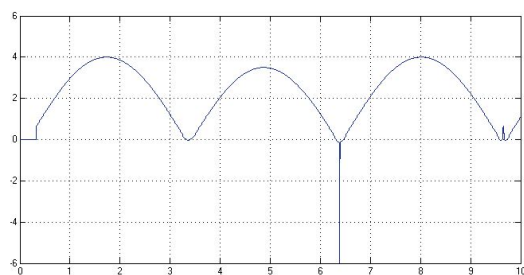


Рис. 59: Думодулированный ЧМ-сигнал

Фазовая автоподстройка частоты — система автоматического регулирования, подстраивающая фазу управляемого генератора так, чтобы она была равна фазе опорного сигнала, либо отличалась на известную функцию от времени. Регулировка осуществляется благодаря наличию отрицательной обратной связи. Выходной сигнал управляемого генератора сравнивается на фазовом детекторе с опорным сигналом, результат сравнения используется для подстройки управляемого генератора.

Система ФАПЧ используется для частотной модуляции и демодуляции, умножения и преобразования частоты, частотной фильтрации, выделения опорного колебания для когерентного детектирования и в других целях.

ФАПЧ сравнивает фазы входного и опорного сигналов и выводит сигнал ошибки, соответствующий разности между этими фазами. Сигнал ошибки проходит далее через фильтр низких частот и используется в качестве управляющего для генератора, управляемого напряжением (ГУН), обеспечивающего отрицательную обратную связь. Если выходная частота отклоняется от опорной, то сигнал ошибки увеличивается, воздействуя на ГУН в сторону уменьшения ошибки. В состоянии равновесия выходной сигнал фиксируется на частоте опорного.

Также в схеме ФАПЧ включен фазовый детектор, который сравнивает фазы двух входных сигналов. Обычно, один из них генерируется генератором сигнала, управляемым напряжением, а второй берется из внешнего источника. ФД имеет два входа, управляющих стоящей за ним схемой подстройки частоты, задача которой сделать фазы сигналов одинаковыми.

Для ФАПЧ сигнал ошибки из ФД (значение найденной разности фаз) подается на сглаживающий фильтр (фильтр нижних частот). Сигнал с фильтра подается на управляемый напряжением генератор, частота(фаза) выходного сигнала которого зависит от напряжения на входе. Сигнал с генератора по цепи обратной связи поступает назад в детектор, замыкая контур ФАПЧ.

Частотная модуляция/демодуляция с фазовой автоподстройкой частоты в Simulink:

В результате были получены следующие характеристики:

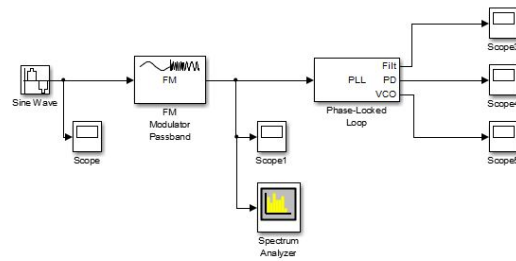


Рис. 60: Частотная модуляция/демодуляция

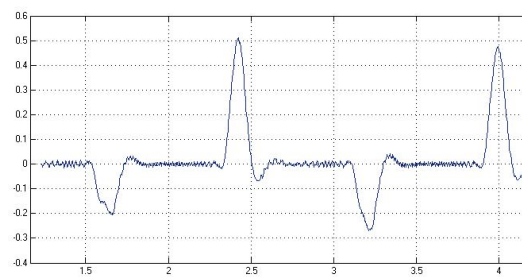


Рис. 61: Выход фильтра

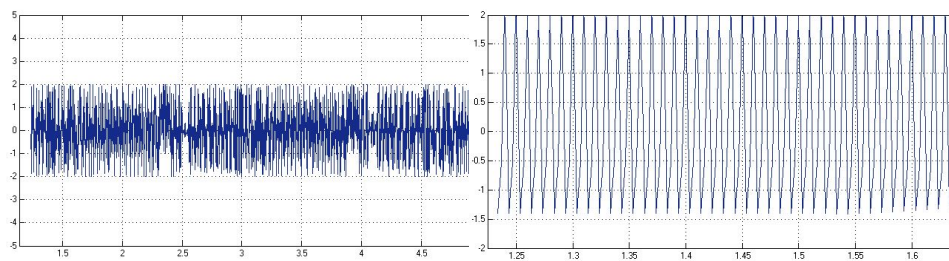


Рис. 62: Выход фазового детектора

Рис. 63: Выход ГУН

## 6.6 Выводы

В результате выполнения данной работы были выполнены частотная и фазовая модуляция/демодуляция, а также частотная демодуляция с помощью блока захвата фазы. Можно сделать вывод, что частотная и фазовая модуляция очень тесно взаимосвязаны, поскольку обе они влияют на аргумент функции  $\cos$ . Поэтому эти два вида модуляции имеют общее название — угловая модуляция. Сигнал с угловой модуляцией имеет вид колебания, начальная фаза которого зависит от времени:

$$s(t) = A_0 \cos(\omega_0 t + j(t)). \quad (15)$$

Различие между фазовой и частотной модуляцией заключается лишь в том, как именно начальная фаза  $j(t)$  связана с модулирующим сигналом.

Для демодуляции использовалась петля ФАПЧ, состоящая из перемножителя (используемого в качестве фазового детектора), фильтра нижних частот и генератора, управляемого напряжением (ГУН). Получаемый на выходе петли ФАПЧ сигнал пропорционален отклонению мгновенной частоты модулированного сигнала от несущей частоты, поэтому при демодуляции ФМ этот сигнал можно дополнительно проинтегрировать, чтобы получить начальную фазу сигнала.