Санкт-Петербургский государственный политехнический университет

Институт компьютерных наук и технологий

Кафедра компьютерных систем и программных технологий

Отчет по лабораторным работам № 1,2

на тему

Сигналы телекоммуникационных систем. Ряд Фурье. Преобразование Фурье. Корреляция.

Работу выполнил:
студент гр. 33501/1
Романов Н.В.

Преподаватель:
Богач Н.В.

Санкт-Петербург
2018 г.

1. Цель работы

Познакомиться со средствами генерации и визуализации простых сигналов.

2. Постановка задачи

1. В командном окне MATLAB и в среде Simulink промоделировать синусоидальный и прямоугольный сигналы с различными параметрами. Получить их спектры. Вывести график.
2. Получить их спектры с помощью преобразования Фурье, вывести на график.
3. Выполнить расчет преобразования Фурье. Перечислить свойства преобразования Фурье.
4. С помощью функции корреляции найдите позицию синхропосылки [101] в сигнале [0001010111000010] . Получите пакет данных, если известно, что его длина составляет 8 бит без учета синхропосылки. Вычислите корреляцию прямым методом, воспользуйтесь алгоритмом быстрой корреляции, сравните время работы обоих алгоритмов

3. Теоретическая информация

Сигнал — материальный носитель информации, используемый для передачи сообщений в системе связи. Сигнал может генерироваться, но его приём не обязателен, в отличие от сообщения, которое рассчитано на принятие принимающей стороной, иначе оно не является сообщением. Сигналом может быть любой физический процесс, параметры которого изменяются (или находятся) в соответствии с передаваемым сообщением.

Спектр сигнала — это совокупность простых составляющих сигнала с определенными амплитудами, частотами и начальными фазами. Между спектром сигнала и его формой существует жесткая взаимосвязь: изменение формы сигнала приводит к изменению его спектра и наоборот, любое изменение спектра сигнала приводит к изменению его формы. Это важно запомнить, поскольку при передаче сигналов в системе передачи, они подвергаются преобразованиям, а значит, происходит преобразование их спектров.

3.1 Ряд и интеграл Фурье

Любая ограниченна, периодическая функция, имеющая конечное число экстремумов на протяжении периода, может быть представлена в виде ряда Фурье:

$$\varphi_p(t) = \sum_{k=-\infty}^{\infty} C_k\, e^{j2\pi k f_1 t}$$

где $f_1 = 1/T_1$; T_1 период функции $\varphi_p(t)$; C_k – постоянные коэффициенты. Коэффициенты могут быть найдены следующим образом:

$$C_k = \frac{1}{T_1}\int_{t_0}^{t_0+T_1} \varphi_p(t) e^{-j\pi k f_1 t} dt$$

При этом значение выражения не зависит от t_0. Обычно берется $t_0=0$ или $t_0=-T_1/2$.

Приведены формулы можно записать в виде одного выражения:

$$\varphi_p(t) = \sum_{k=-\infty}^{\infty} \left[\frac{1}{T_1}\int_{t_0}^{t_0+T_1} \varphi_p(t) e^{-j\pi k f_1 t} dt\right] e^{j2\pi k f_1 t}$$

Ряд Фурье справедлив для периодических сигналов, однако на его основе можно вывести соотношения и для непериодических сигналов. В этом случае T1 стремится к бесконечности, в связи с этим частота f1 стремится к нулю и обозначается как *df, kf1* является текущим значением частоты f, а сумма меняется на интеграл. В результате получается выражение:

$$\varphi_p(t) = \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} \varphi_p(t) e^{-j2\pi f t} dt\right] \varphi_p(t) e^{-j2\pi f t} df$$

Это выражение называется интегралом Фурье и объединяет прямое преобразование Фурье:

$$\Phi(f) = \int_{-\infty}^{\infty} \varphi_p(t) e^{-j2\pi f t} dt$$

и обратное преобразование Фурье:

$$\varphi(t) = \int_{-\infty}^{\infty} \Phi_p(t) e^{j2\pi f t} dt$$

Приведенные преобразования существуют только для функций с ограниченной энергией:

$$\int_{-\infty}^{\infty} |\varphi(t)|^2 dt \neq \infty$$

В большинстве случаев термин преобразование Фурье обозначает именно интеграл Фурье. Преобразование Фурье сигнала так же называется спектром сигнала.

3.2 Свойства преобразования Фурье

Преобразование Фурье имеет ряд свойств:

- Суммирование функций

Преобразование Фурье – линейное преобразование. Отсюда следует:

$$\sum_{i=1}^{n} \alpha_i \varphi_i(t) \leftrightarrow \sum_{i=1}^{n} \alpha_i \Phi_i(f)$$

где α_i постоянный коэффициент.

- Смещение функций

При смещении функции t_0 ее ПФ умножается на $e^{j2\pi f t_0}$

$$\varphi(t - t_{0)} \leftrightarrow e^{-j2\pi f t_0} \Phi(f)$$

- Изменение масштаба аргумента функции

При домножении аргумента функции t на постоянный коэффициент α, ПФ функции имеет вид $\frac{1}{|\alpha|}\Phi(\frac{f}{\alpha})$:

$$\varphi(\alpha t) \leftrightarrow \frac{1}{\alpha}\,\Phi(\frac{f}{\alpha})$$

- Перемножение функций

ПФ произведения двух функции равно свертки ПФ этих функций:

$$\varphi_1(t)\varphi_2(t) \leftrightarrow \Phi_1(f) * \Phi_1(f)$$

- Свертывание функций

ПФ свертки двух функций равно произведению ПФ этих функций:

$$\varphi_1(t)\varphi_2(t) \leftrightarrow \Phi_1(f)\Phi_1(f)$$

- Дифференцирование функции

При дифференцировании функции ее ПФ домножается на $j2\pi f$:

$$\frac{d[\varphi(t)]}{dt} \leftrightarrow j2\pi f\Phi(t)$$

- Интегрирование функции

При интегрировании функции ее ПФ делится на $j2\pi f$:

$$\int_{-\infty}^{t} \varphi(t')dt' \leftrightarrow \frac{1}{j2\pi f}\Phi(t)$$

- Обратимость преобразования

Преобразование обратимо с точностью до знака аргумента.

3.3 Корреляция

Для нахождения посылки в сигнале можно использовать алгоритм взаимной корреляции, где N – длина всех x и y. Для нахождения посылки можно сдвигать один вектор относительно другого, каждый раз находя значение корреляции. Максимальная корреляция будет соответствовать месту искомой посылки:

$$R = \frac{1}{N}\sum_{i=1}^{N} x_i * y_i$$

Алгоритм быстрой корреляции:

$$R = \frac{1}{N}F_d^{-1}[X_s' * Y]$$

4. Ход работы

4.1 Синусоидальный сигнал в MATLAB

Для задания синусоидального сигнала в среде Matlab использована стандартная функция sin, которая в качестве аргументов получает частоту сигнала с необходимыми коэффициентами, время наблюдения сигнала и заданный фазовый сдвиг. Частота f = 25 Гц, фазовый сдвиг f0 = 3, время наблюдения сигнала t – от 0 до 200 секунд, время начала преобразования t1=0. Код программы в среде Matlab представлен ниже:

```
f = 25;
f0 = 3;
t1=0;
t=0:1:200;
s = 2*sin(3.01*f*t+f0)
plot(t, s);
figure;
dots = 1024;
fft(s,dots);
plot(abs(fft(s, dots)))
```

В результате выполнения данной программы получили чистый синусоидальный сигнал с заданными параметрами.

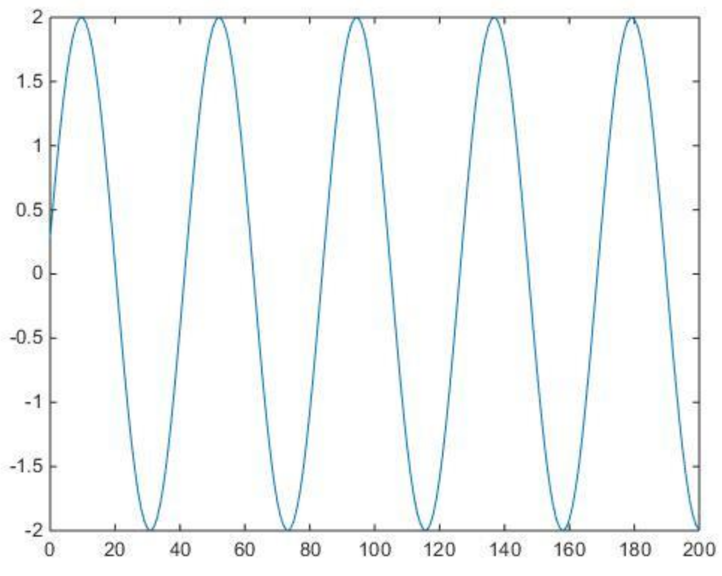

Рисунок 1 Синусоидальный сигнал 1

Для получения спектра сигнала была использована стандартная функция среды Matlab – fft, которая реализует БПФ. В качестве аргументов у данной функции выступает сам заданный сигнал и количество используемый точек для быстрого преобразования Фурье. При использовании данной функции был получен следующий спектр сигнала:

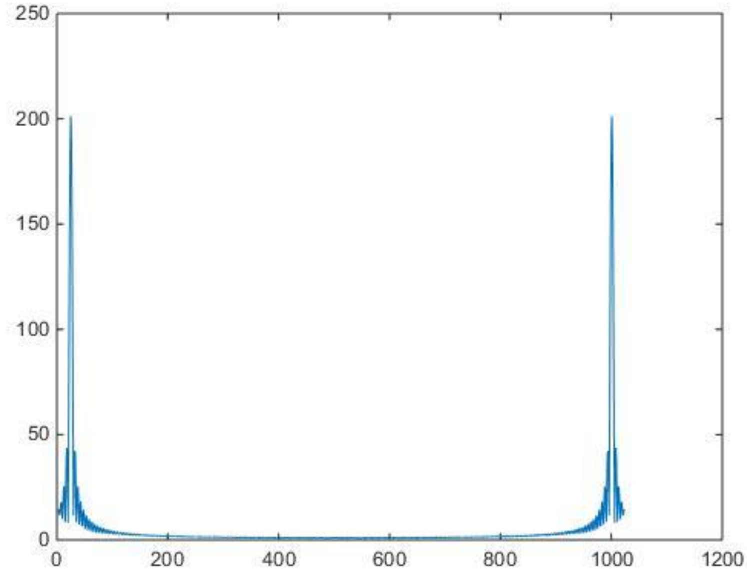

Рисунок 2 Спектр синусоидального сигнала 1

При помощи ограничения наблюдаемой области спектра на графике (xlim), рассмотрим данный спектр ближе, что позволит нам обнаружить предполагаемые в теории «колокольчики» у спектра синусоидального сигнала.

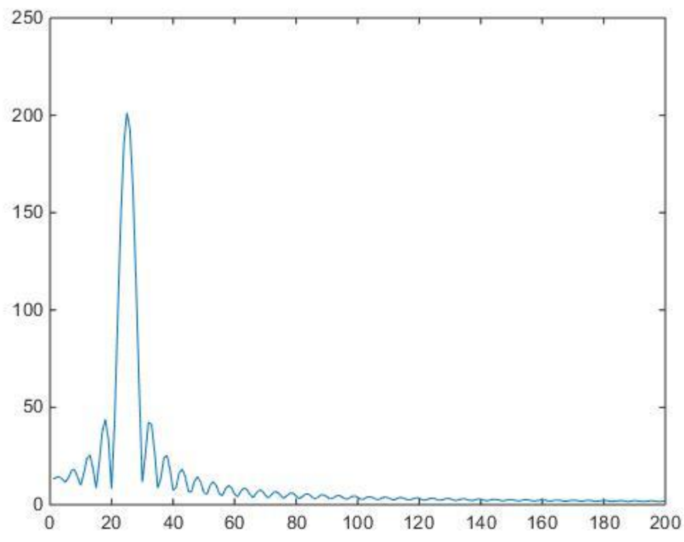

Рисунок 3 Приближенный спектр заданного синусоидального сигнала 1

Изменим коэффициент перед частотой заданного сигнала и уменьшим время моделирования, чтобы рассмотреть спектр полученного сигнала.

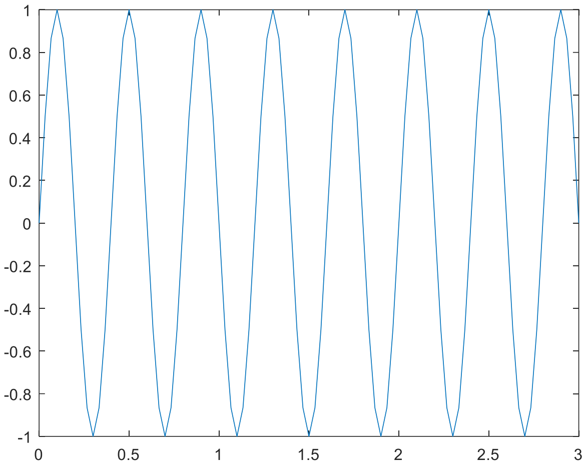

Рисунок 4 Синусоидальный сигнал 2

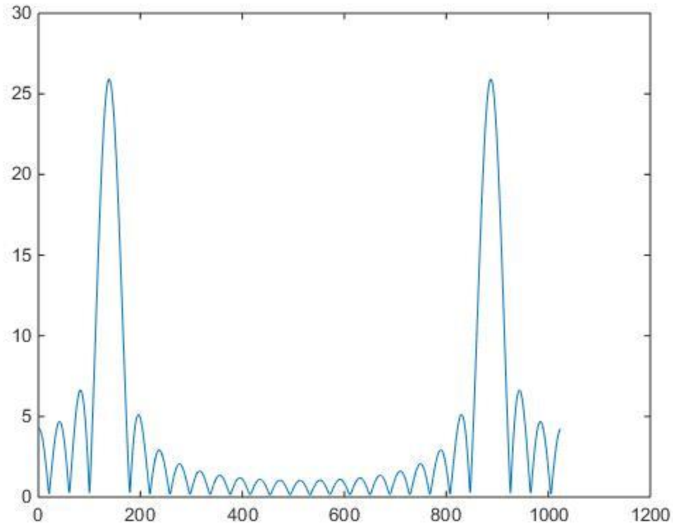

Рисунок 5 Спектр синусоидального сигнала 2

4.2 Прямоугольный сигнал в MATLAB

Для задания прямоугольного сигнала в среде Matlab была использована стандартная функция: square(t,duty), где t – время наблюдения сигнала, duty – скважность заданного сигнала. Код программы приведен ниже:

```
t=0:.1:20;
duty = 70;
x = square(t,duty)
plot(t, x);
ylim([-1.1 1.1]);
figure;
dots = 2500;
fft(x,dots);
plot(abs(fft(x, dots)))
%xlim([-1 300]);
```

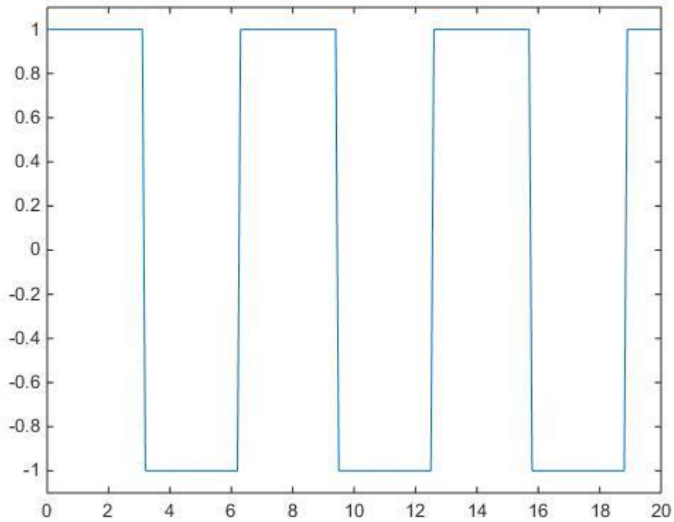

Рисунок 6 Прямоугольный сигнал 1

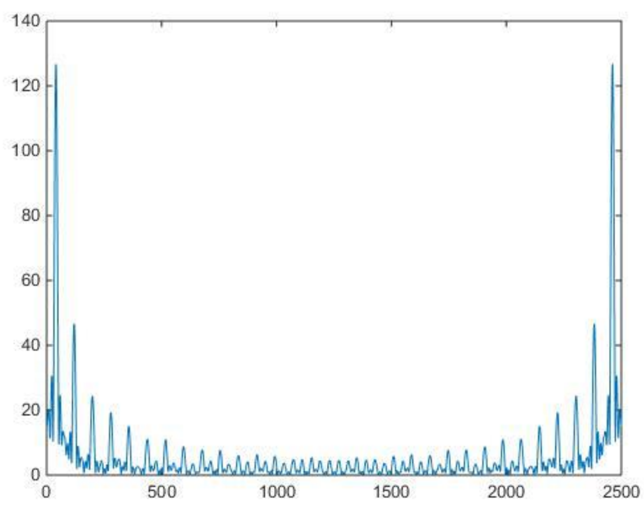

Рисунок 7 Спектр прямоугольного сигнала 1

Изменим скважность прямоугольного сигнала и получим его график и спектр:

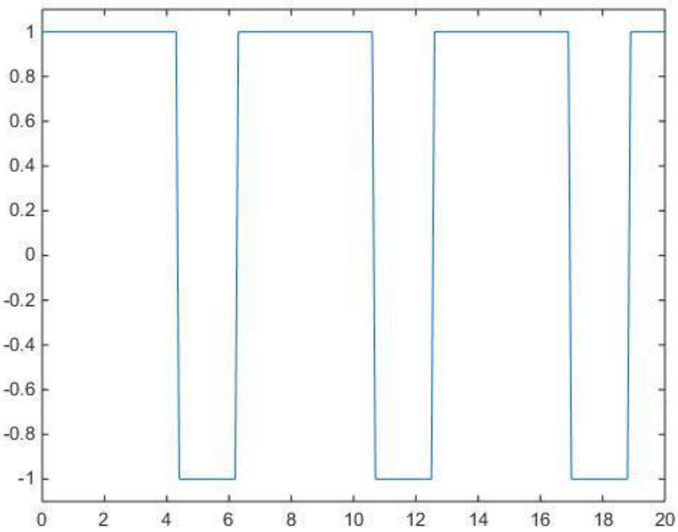

Рисунок 8 Прямоугольный сигнал 2

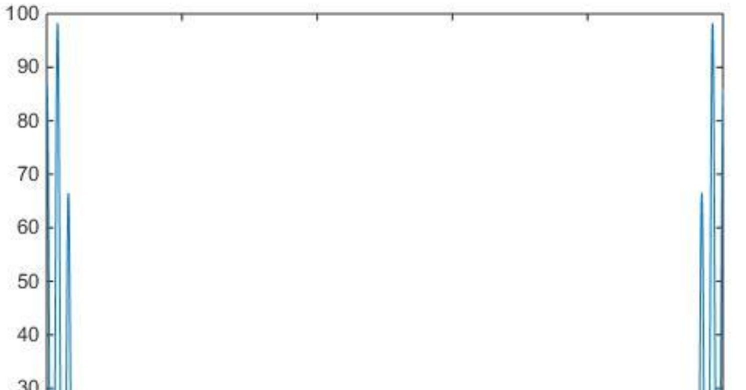

Рисунок 9 Спектр прямоугольного сигнала 2

4.3 Синусоидальный сигнал в Simulink

Проделаем аналогичные действия в среде разработки Simulink. Будем использовать стандартные примитивы из библиотеки: Scope (для наблюдения за сигналов), Spectrum Analyzer (для вывода графика спектра сигнала), Sine Wave (для задания синусоидального сигнала), Pulse Generator (для задания прямоугольного сигнала).

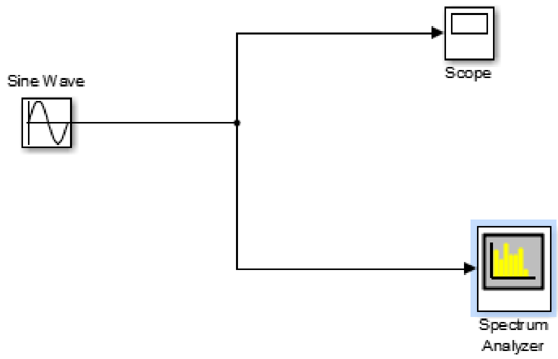

Рисунок 10 Схема для синусоидального сигнала

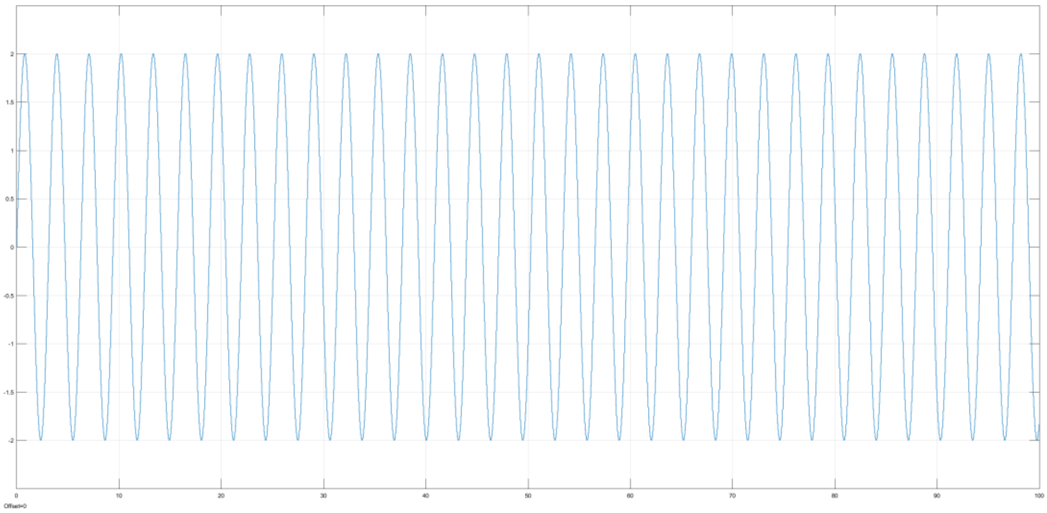

Рисунок 31 Синусоидальный сигнал 1 в Simulink

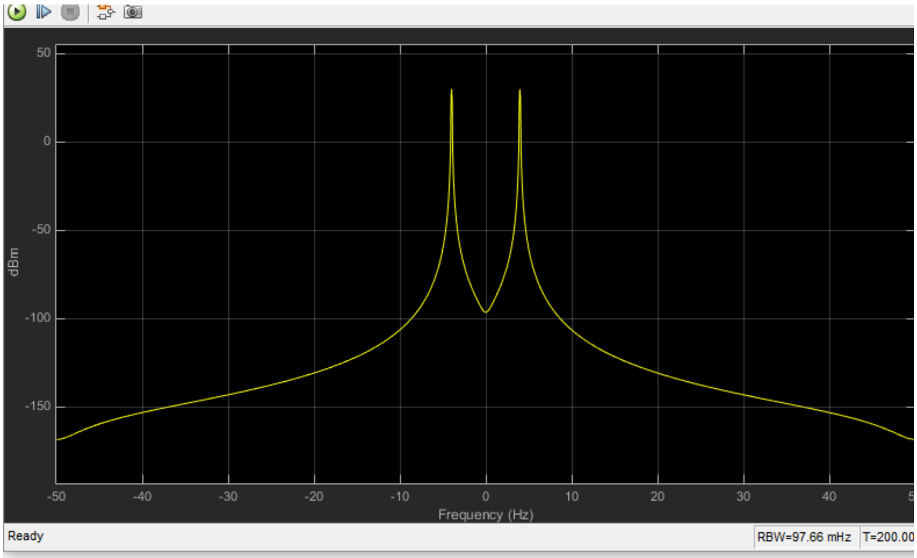

Рисунок 42 Спектр синусоидального сигнала 1 в Simulink

Изменим частоту сигнала и получим его спектр:

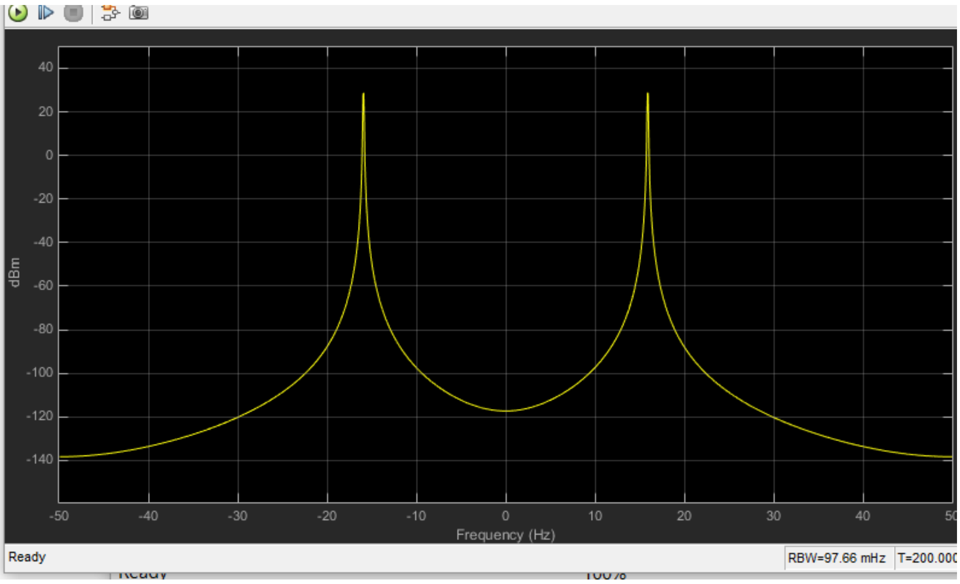

Рисунок 5 Спектр синусоидального сигнала 2 в Simulink

4.4 Прямоугольный сигнал в Simulink

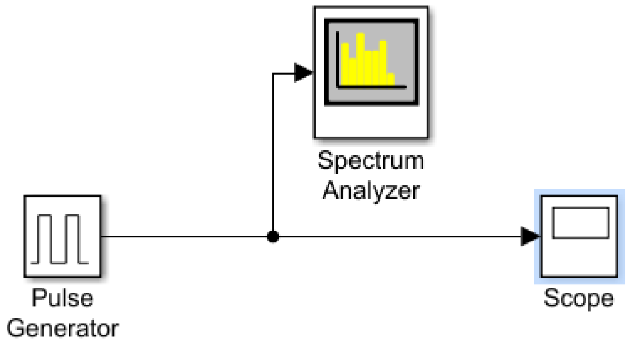

Рисунок 6 Схема для прямоугольного сигнала

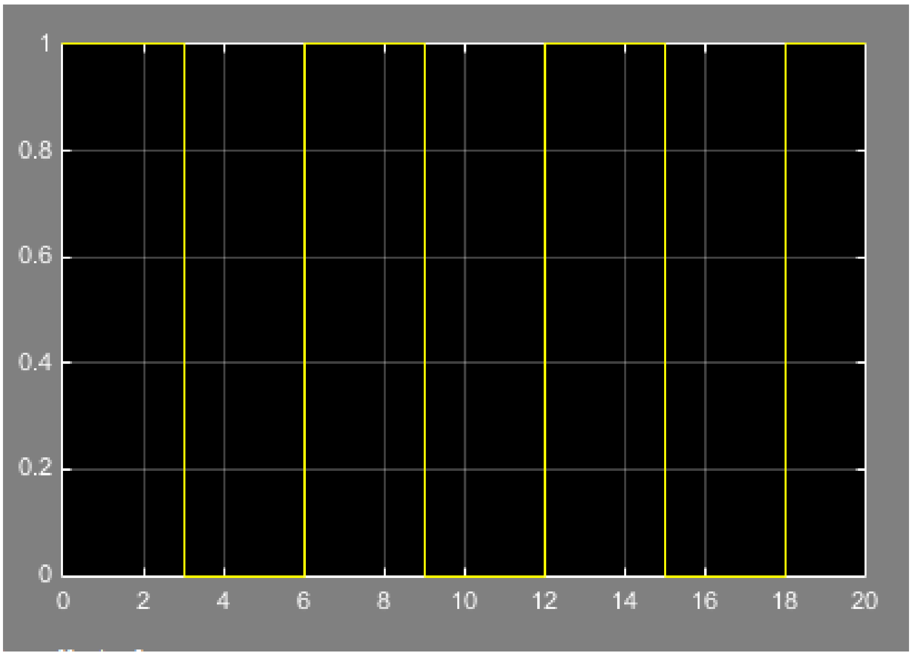

Рисунок 7 Прямоугольный сигнал 1

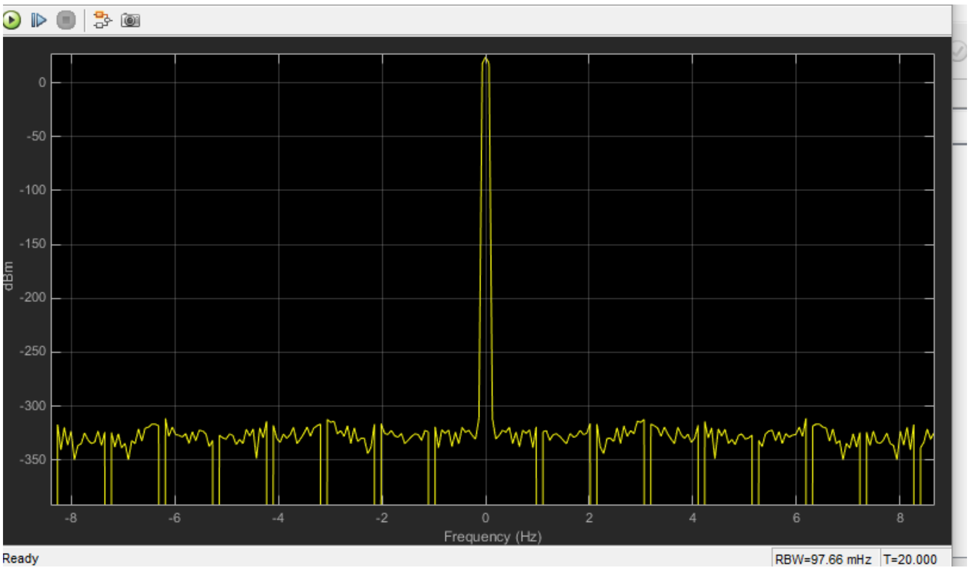

Рисунок 8 Спектр прямоугольного сигнала 1

Построим измененный прямоугольный сигнал и получим его спектр:

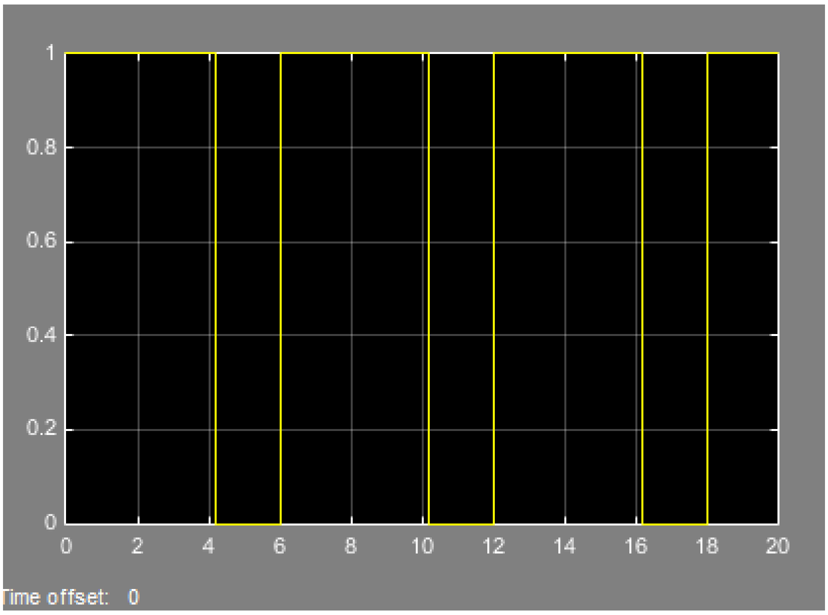

Рисунок 9 Прямоугольный сигнал 2

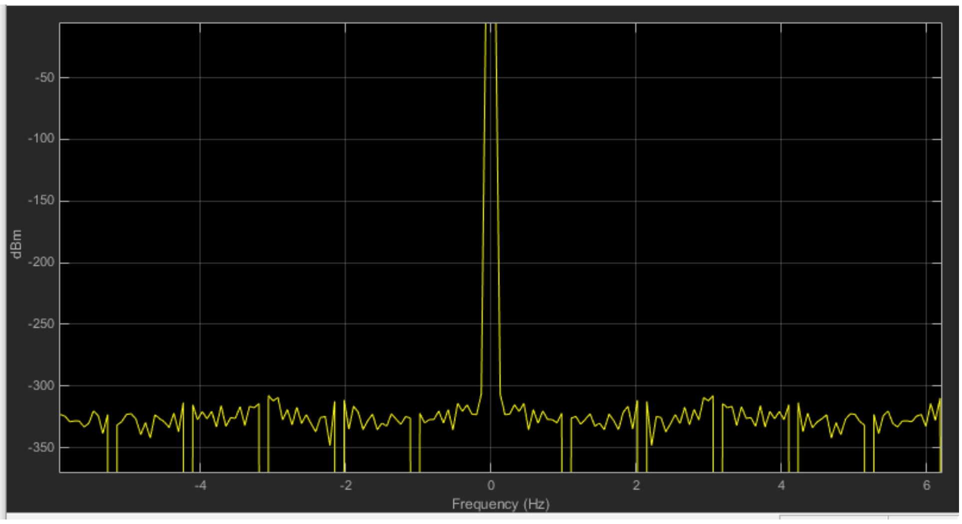

Рисунок 10 Спектр прямоугольного сигнала 2

4.5 Расчет преобразования Фурье

В пунктах 4.1 – 4.4 рассматривались синусоидальные и прямоугольные сигналы. Применим преобразование Фурье для синусоидальной и прямоугольной функций:

```
sin_f  = A*sin(2*pi*f*t + ph) + b;
f_ft = fourier(sin_f)

pryam_f = A*rectangularPulse(-T, T, t);
f_ft1 = fourier(pryam_f)
```

где A – амплитуда, f – частота, ph – фаза и b – смещение.
В результате получим две формулы:

```
f_ft =2*pi*b*dirac(w) - A*pi*(dirac(w - 2*pi*f)*exp(ph*1i) - dirac(w + 2*pi*f)*exp(-ph*1i))*1i


f_ft1 =A*((sin(T*w) + cos(T*w)*1i)/w - (cos(T*w)*1i - sin(T*w))/w)
```

4.6 Корреляция

```
pos = [0 0 0 1 0 1 0 1 1 1 0 0 0 0 1 0];
sp = [1 0 1];
sp_d = zeros (1, length(pos));
sp_d( sp == 1) = 1;
sp_d( sp == 0 ) = -1;
pos_t = pos;
R = zeros(1, length(pos_t));
tic % прямой метод
for i = 1:length(pos_t)
    R(i) = sum(pos_t.*circshift(sp_d, i-1, 2))/length(pos_t);
end
toc
tic % быстрый метод
pos_t = fft(pos_t);
sp_d = fft(sp_d);
pos_t = conj(pos_t);
```

```
BR=ifft(pos_t.*sp_d)/length(pos_t);
Toc
```

В качестве исходного примера возьмем синхропосылку 101 в сигнале 0001010111000010.

До начала вычисления корреляции изменим синхропосылку с 1 0 1 на 1 -1 1, что обеспечит более точное ее нахождение в посылке, также дополним ее нулями для совпадения длин векторов. Произведем два расчета корреляции – обычным алгоритмом и быстрым с контролем времени на каждую операцию.

Первый алгоритм показал время выполнения - 0.045 мс, а второй – 0.0020.

Можно сделать вывод, что алгоритм быстрой корреляции эффективней обычного. Разница не значительна из-за малой длины посылки.

Стоит отметить, что на таком коротком примере алгоритм быстрой корреляции оказался всего в 2 раза быстрее обычного алгоритма. Его эффективность вырастет во много раз, по сравнению с обычным алгоритмом на больших посылках.

5. Вывод

В данной лабораторной работе нами были получены навыки работы со средствами генерации и построения графиков синусоидальных сигналов и прямоугольных импульсов. Для осуществления поставленных задач были использованы среды разработки MatLab и Simulink. В ходе работы были проанализированы спектры заданных сигналов.

Во второй части лабораторной работы был получен опыт по применению преобразований Фурье. Данное преобразование используется в телекоммуникационных технологиях для обработки различных данных (возможно сжатие, кодирование, декодирование данных и т.д.). Так же преобразование Фурье часто используется в алгоритмах, осуществляющих фильтрацию сигналов, их модуляцию или демодуляцию.