Санкт-Петербургский политехнический университет Петра Великого
Кафедра компьютерных систем и программных технологий

Отчёт по лабораторной работе

Дисциплина: Телекоммуникационные технологии

Тема: Сигналы телекоммуникационных систем.
Преобразование Фурье. Корреляция

Выполнил студент гр. 33501/4 Мальцев М.С.
Преподаватель Богач Н.В.

Санкт-Петербург
15 апреля 2018 г.

0 Содержание

1 Цель работы

Познакомиться со средствами генерации и визуализации простых сигналов. Получить представление о спектрах телекоммуникационных сигналов.

2 Постановка задачи

- В командном окне MATLAB и в среде Simulink промоделировать синусоидальный и прямоугольный сигналы с различными параметрами. Получить их спектры. Вывести на график.
- Выполнить расчет преобразования Фурье. Перечислить свойства преобразования Фурье.
- С помощью функции корреляции найти позицию синхропосылки [101] в сигнале [0001010111000010]. Получить пакет данных, если известно, что его длина составляет 8 бит без учета синхропосылки. Вычислить корреляцию прямым методом, используя алгоритмом быстрой корреляции, сравнить время работы обоих алгоритмов.
- Быстрая корреляция

3 Теоретический раздел

3.1 Сигналы

Сигнал – это физический процесс, который несёт некоторую информацию.

Классификация сигналов:

1. По физической природе носителя информации:
   - электрические
   - электромагнитные
   - оптические
   - акустические
   - и другие
2. По способу задания сигнала:
   - детерминированные (описываемые аналитической функцией)

- случайные (для их описания используется аппарат теории вероятностей)

3. непрерывные и дискретные

4. периодические и непериодические

5. бесконечные и конечные

3.2 Преобразования Фурье

Преобразования Фурье осуществляется с помощью ряда Фурье и с помощью интеграла Фурье, причём первый применяется когда функция периодическая, а второй когда она апериодична.

Ряд Фурье – представление функции f с периодом τ в виде ряда:

$$f(x) = \frac{a_0}{2} + \sum_{k=1}^{+\infty} A_k cos(k\frac{2\pi}{\tau}x + \theta_k) \tag{3.1}$$

Интегралы Фурье имеют вид:

$$F(j\omega) = \int_{-\infty}^{\infty} f(t)e^{-it\omega}dt \tag{3.2}$$

$$f(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} f(j\omega)e^{-it\omega}d\omega \tag{3.3}$$

Этот метод может применяться только для абсолютно интегрируемых функций времени, удовлетворяющих неравенству:

$$\int_{-\infty}^{\infty} f(t)^2 dt < \infty \tag{3.4}$$

3.3 Корреляция сигналов

Корреляция является методом анализа сигналов. В качестве примера использования метода можно привести следующее, допустим, что имеется сигнал s(t), в котором может быть (а может и не быть) некоторая последовательность x(t) конечной длины T, временное положение которой нас интересует. Для поиска этой последовательности в скользящем по сигналу s(t) временном окне длиной T вычисляются скалярные произведения сигналов s(t) и x(t). Тем самым мы "прикладываем"искомый сигнал x(t) к сигналу s(t), скользя по его аргументу, и по величине скалярного произведения оцениваем степень сходства сигналов в точках сравнения. Для сигналов $s(t)$ и $x(t)$ кросс-корреляция будет вычисляться по формуле 3.5.

$$\omega(t) = s(t) \otimes x(t) \triangleq \int_{-\infty}^{\infty} s^*(\tau)\ x(\tau + t)\ d\tau \tag{3.5}$$

Корреляционный анализ дает возможность установить в сигналах (или в рядах цифровых данных сигналов) наличие определенной связи изменения значений сигналов по независимой переменной, то есть, когда большие значения одного сигнала (относительно средних значений сигнала) связаны с большими значениями другого сигнала (положительная корреляция), или, наоборот, малые значения одного сигнала связаны с большими значениями другого (отрицательная корреляция), или данные двух сигналов никак не связаны (нулевая корреляция).

В варианте автокорреляции (autocorrelation) по аналогичной методике производится определение скалярного произведения сигнала s(t) с собственной копией, скользящей по аргументу. Автокорреляция позволяет оценить среднестатистическую зависимость текущих отсчетов сигнала от своих предыдущих и последующих значений (так называемый радиус корреляции значений сигнала), а также выявить в сигнале наличие периодически повторяющихся элементов.

Особое значение методы корреляции имеют при анализе случайных процессов для выявления неслучайных составляющих и оценки неслучайных параметров этих процессов.

Автокорреляционная функция (АКФ) сигнала s(t), конечного по энергии, является количественной интегральной характеристикой формы сигнала, выявления в сигнале характера и параметров взаимной временной связи отсчетов, что всегда имеет место для периодических сигналов, а также интервала и степени зависимости значений отсчетов в текущие моменты времени от предыстории текущего момента. АКФ определяется интегралом от произведения двух копий сигнала s(t), сдвинутых относительно друг друга на время τ:

$$\omega(t) \triangleq \int_{-\infty}^{\infty} s^*(\tau)\ s(\tau + t)\ d\tau = ||s(\tau)||\ ||s(\tau + t)||\ cos\varphi(\tau) \tag{3.6}$$

4 Ход работы

4.1 Моделирование синусоидального сигнала

4.1.1 Получение непрерывного сигнала

При открытие Simulink был выбран шаблон Simple Simulation.

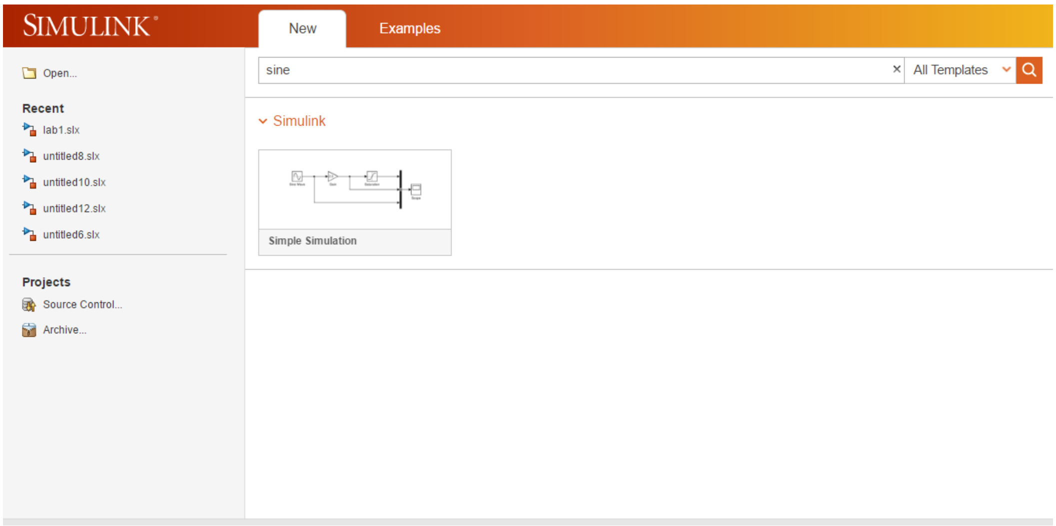


Рис. 4.1: Выбор шаблона в начальном окне Simulink.

Была сгенерирована схема представленная на рисунке 4.2.

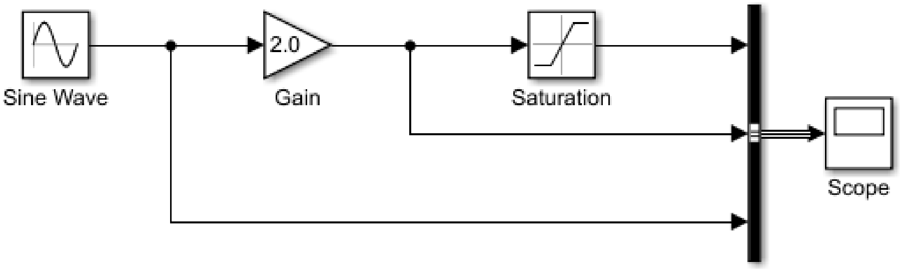


Рис. 4.2: Схема автоматически сгенерированная Simulink.

Краткое описание назначения элементов:

- **Sine Wave** задаёт синусоидальный сигнал с амплитудой 1 и частотой 1 rad/sec
- **Gain** усиливает входной сигнал в 2 раза
- **Saturation** устанавливает ограничивающие пределы верхний на 0.5 и нижний на -0.5

Таким образом, при симуляции мы должны увидеть на графике 3 сигнала:

1. синусоидальный сигнал с амплитудой 1
2. синусоидальный сигнал с амплитудой 2

3. сигнал трапециевидной формы с амплитудой 0.5

Причём, для всех сигналов должен быть одинаковый период, равный ~ 6.28 секунды.

При запуске симуляции получили результаты продемонстрированные на рисунке 4.3.

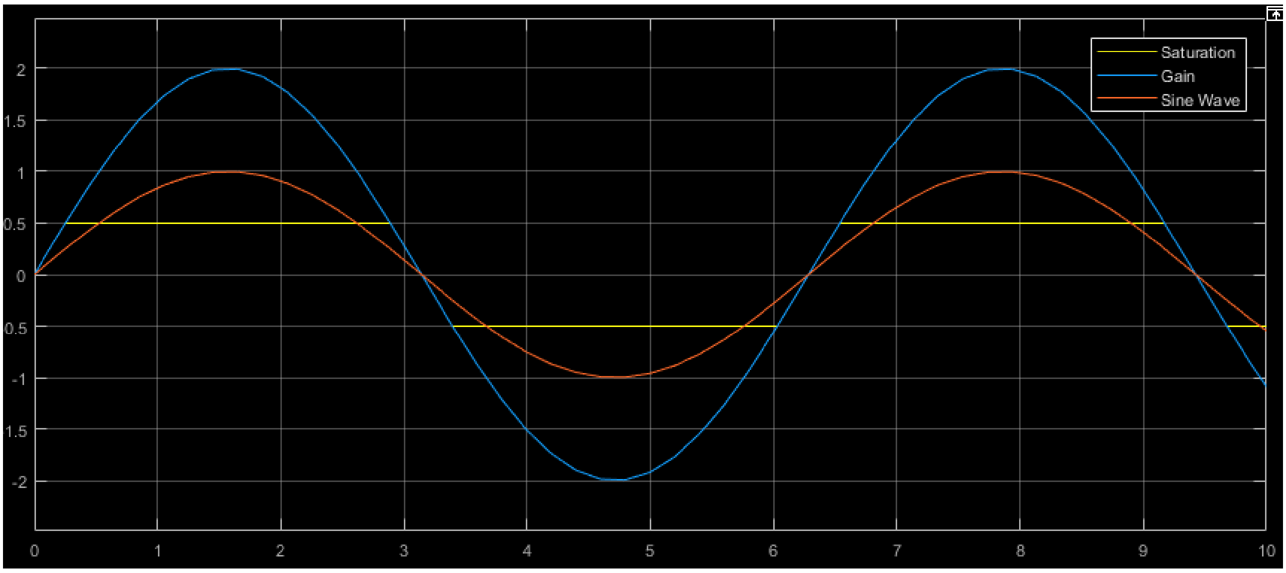


Рис. 4.3: Результат симуляция непрерывного сигнала. Окно Scope.

Проанализировав результаты симуляции, на соответствие ожиданиям, можно сделать вывод, что она выполнена правильно.

4.1.2 Получение дискретного сигнала

Не изменяя общую структуру, представленную на рисунке 4.2, изменим для элемента **Sine Wave** параметр *Sine type* с *Time based* на *Sample based*, таким образом мы сделаем сигнал дискретным. Установим *Samples per period* на 20π, *Sample time* на 0.1.

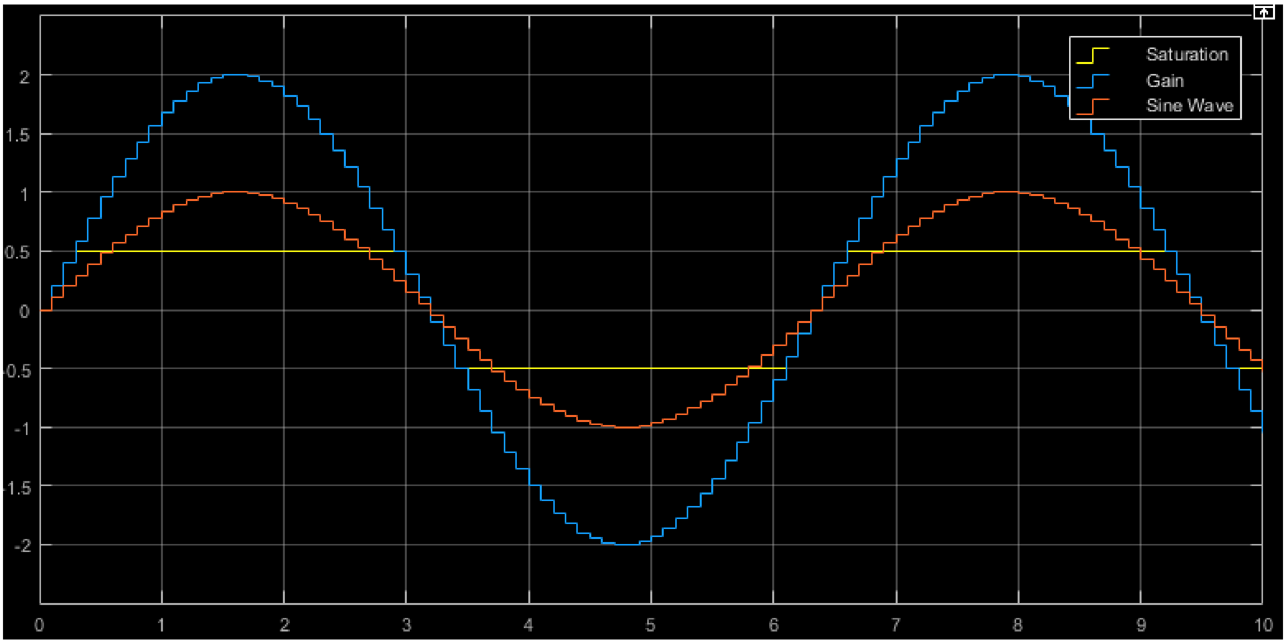


Рис. 4.4: Результат симуляция дискретного сигнала. Окно Scope.

На рисунке 4.4 видно, что непрерывный сигнал стал дискретным, что соответствует нашим ожиданиям.

4.1.3 Получение спектра дискретного сигнала

Для дискретного сигнала получим его спектр. Для этого установим *Sample time* на 0.01 и *Simulation stop time* на 20.

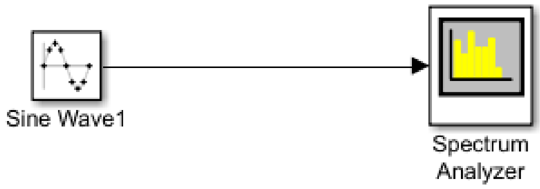


Рис. 4.5: Схема для исследования спектра дискретного синусоидального сигнала.

При запуске симуляции был получен результат, продемонстрированный на рисунке 4.6.

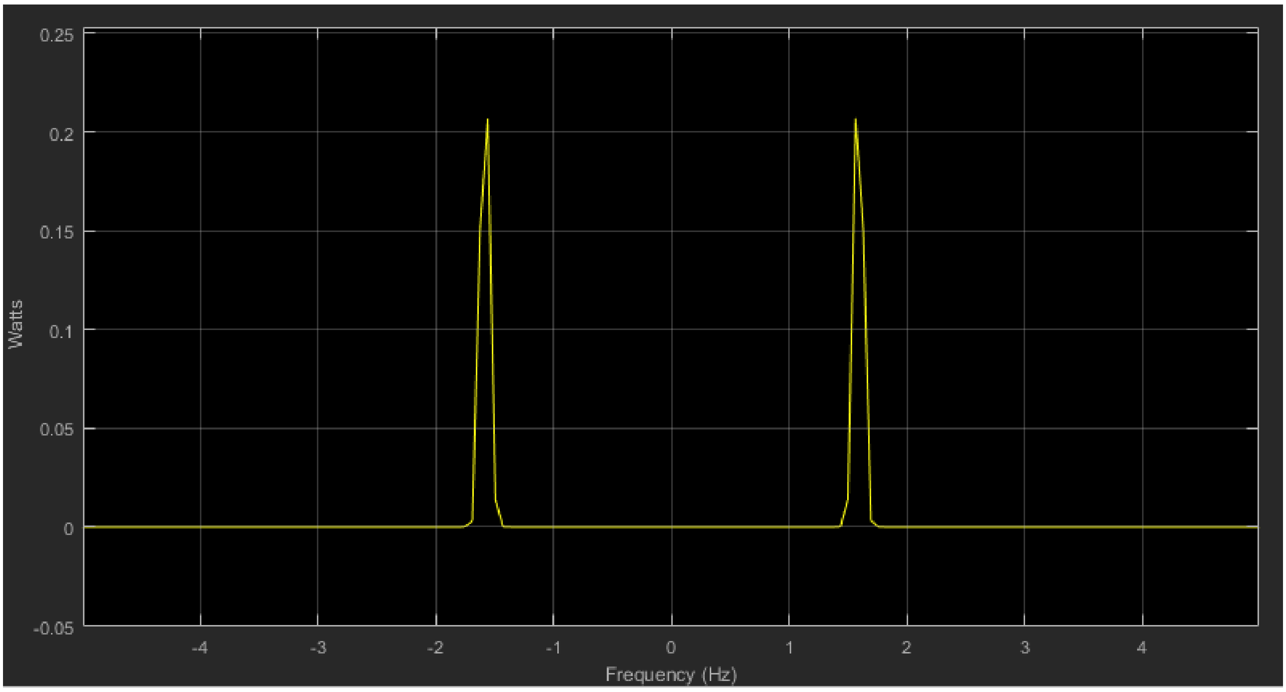


Рис. 4.6: Спектр синусоидального дискретного сигнала. Окно Spectrum Analyzer.

Изменим амплитуду входного сигнала с 1 до 5 и промодулируем снова.

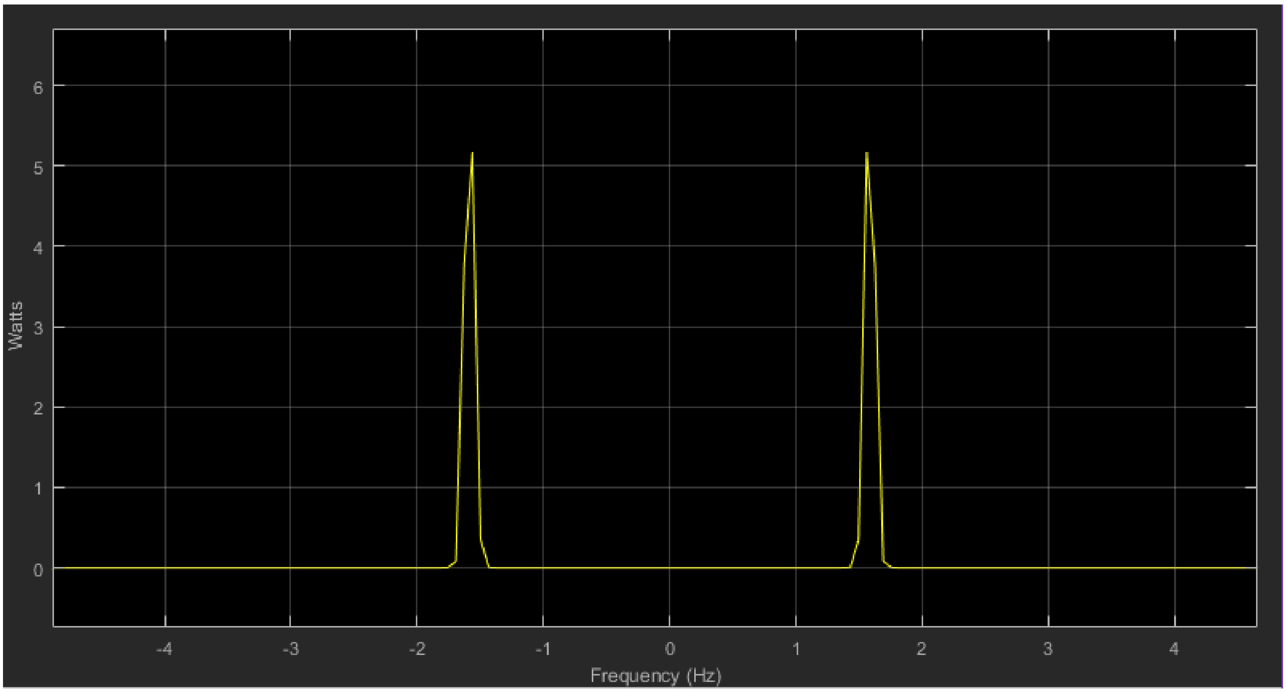


Рис. 4.7: Спектр синусоидального дискретного сигнала. Окно Spectrum Analyzer.

Изменим *Samples per period* с 20π до 40π

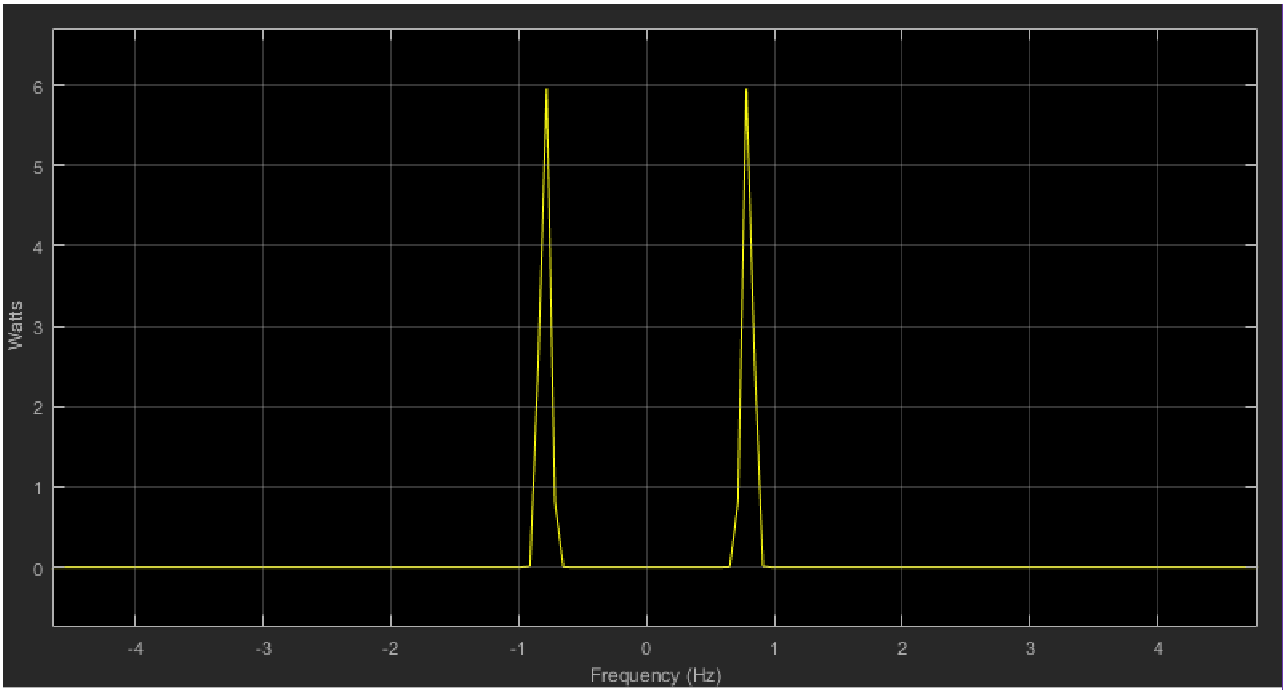


Рис. 4.8: Спектр синусоидального дискретного сигнала. Окно Spectrum Analyzer.

По полученным результатам варьирования параметров задания сигнала можно сделать вывод о том, что моделирование было проведено верно. На рисунках 4.6, 4.7, 4.8 продемонстрировано, что при изменение амплитуды сигнала изменяется амплитуда спектра, причём нелинейно, а при изменение периода обратно пропорционально изменяется частота спектра.

4.2 Моделирование прямоугольного сигнала

4.2.1 Получение дискретного сигнала

Для исследования прямоугольного дискретного сигнала была введена схема представленная на рисунке 4.9. *Simulation stop time* установлен на 20.

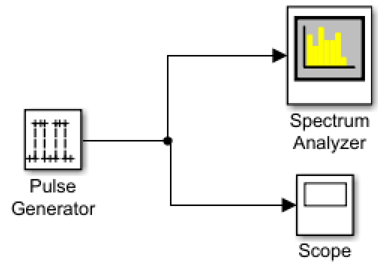


Рис. 4.9: Схема для исследования прямоугольного дискретного сигнала

Для **Pulse Generator** были заданы параметры представленные на рисунке 4.10

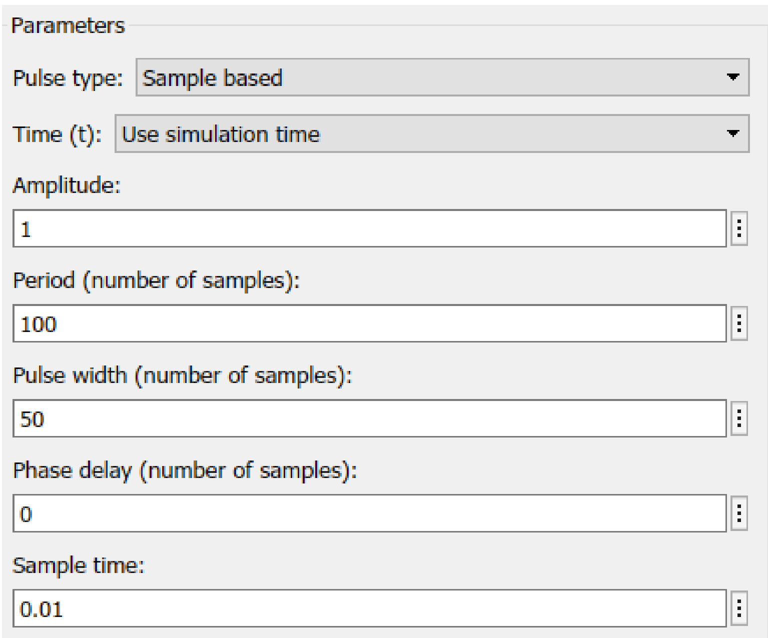


Рис. 4.10: Окно Block Parameters: Pulse Generator. Раздел Parameters.

После моделирования в окне Scope были получены результаты, продемонстрированные на рисунке 4.11.

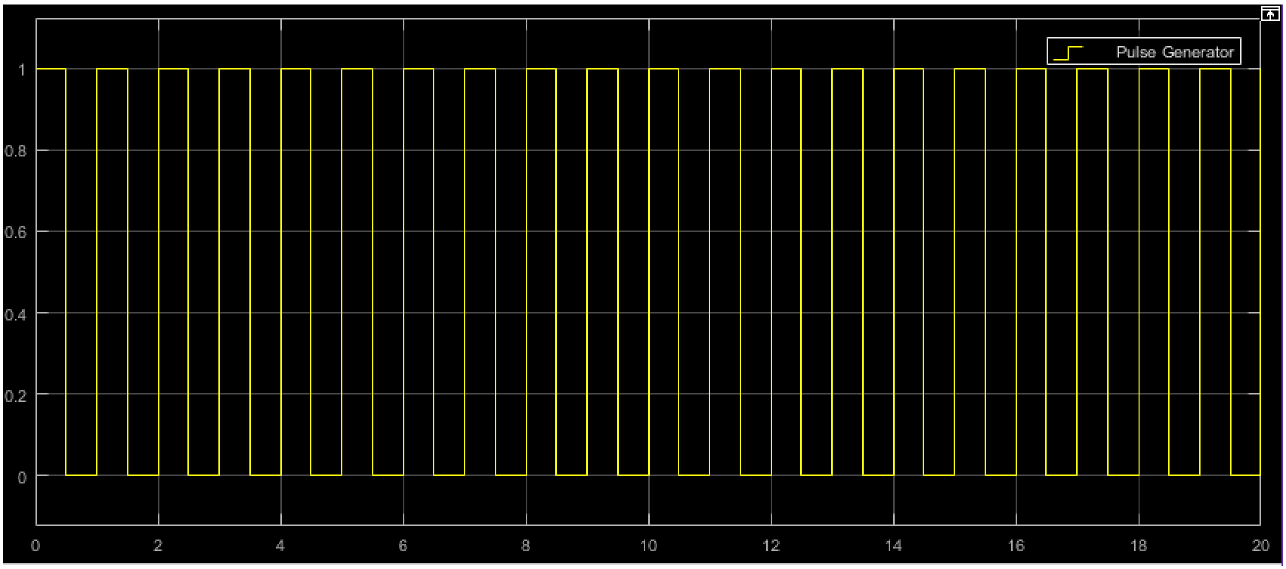


Рис. 4.11: Результаты симуляции дискретного прямоугольного сигнала. Окно Scope.

По результатам симуляции, визуально можно определить, что поставленная задача, смоделировать прямоугольный сигнал, выполнена.

4.2.2 Получение спектра дискретного сигнала

Для получения спектра дискретного сигнала воспользуемся схемой приведённой на рисунке 4.9.

Результаты проведённой симуляции приведены на рисунке 4.12.

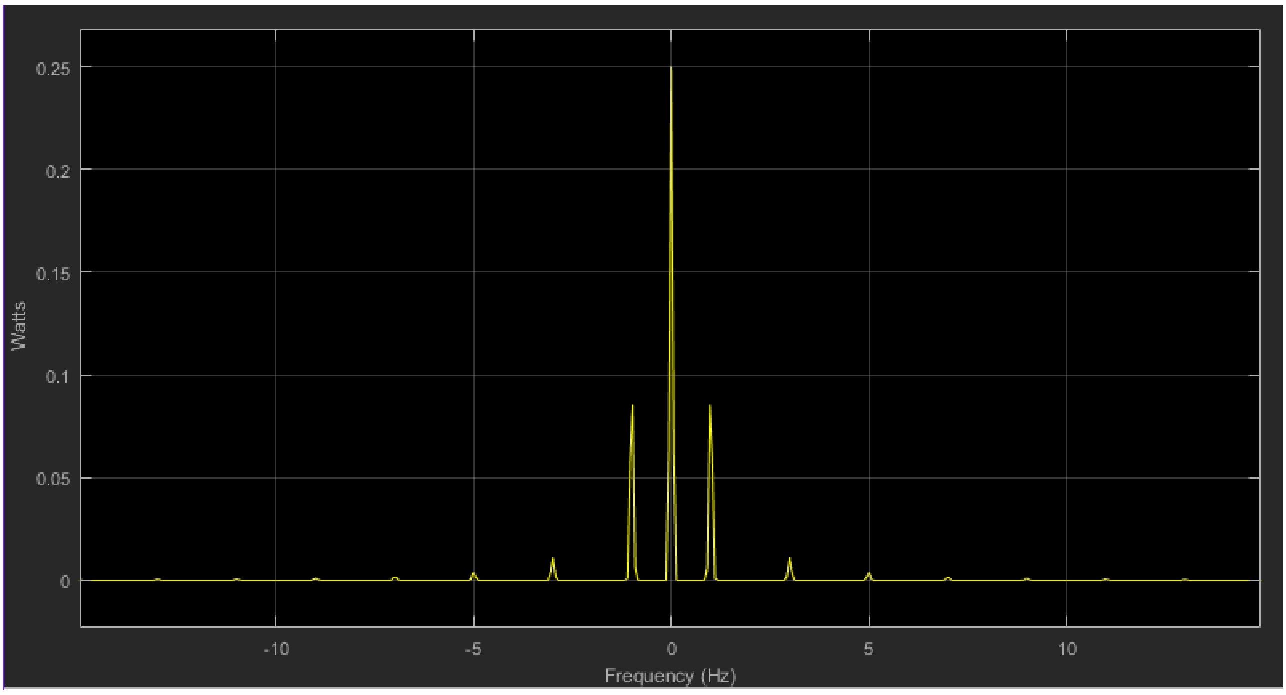


Рис. 4.12: Полученный спектр для дискретного прямоугольного сигнала. Окно Spectrum Analyzer.

Будем изменять параметры сигнала и следить за изменением спектра. Изменим период сигнала со 100 до 75.

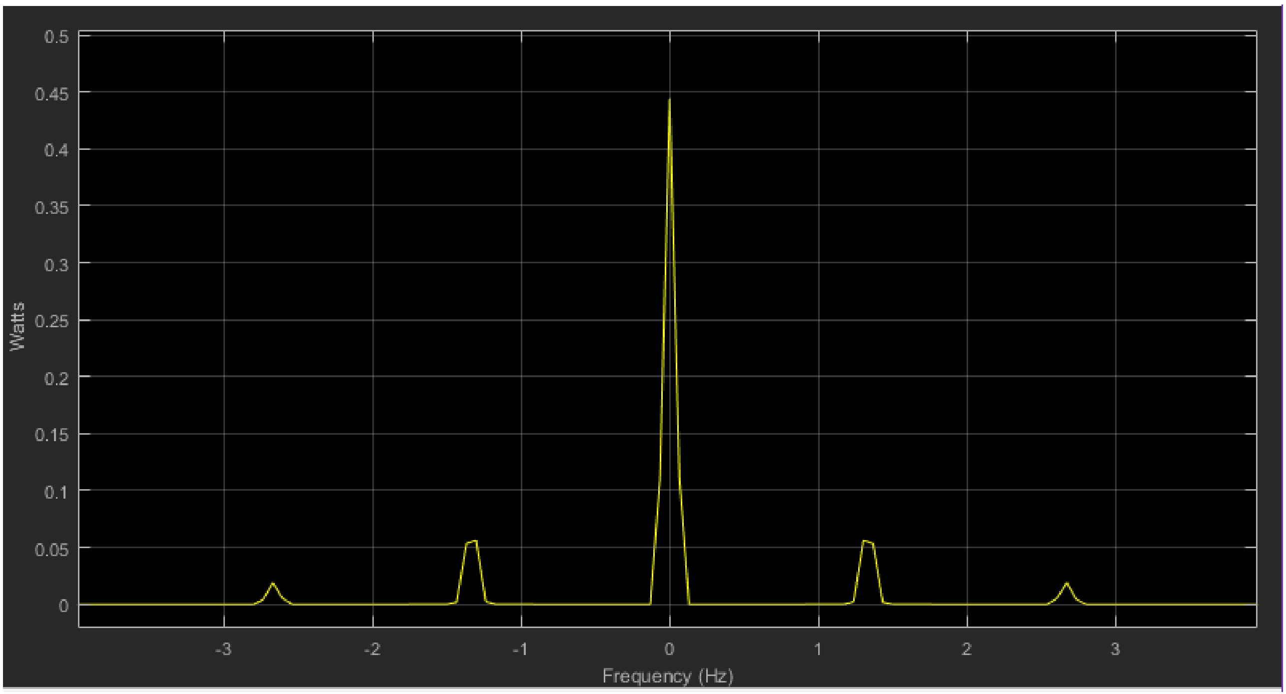


Рис. 4.13: Полученный спектр для дискретного прямоугольного сигнала. Окно Spectrum Analyzer.

Изменим длину импульса с 50 до 25.

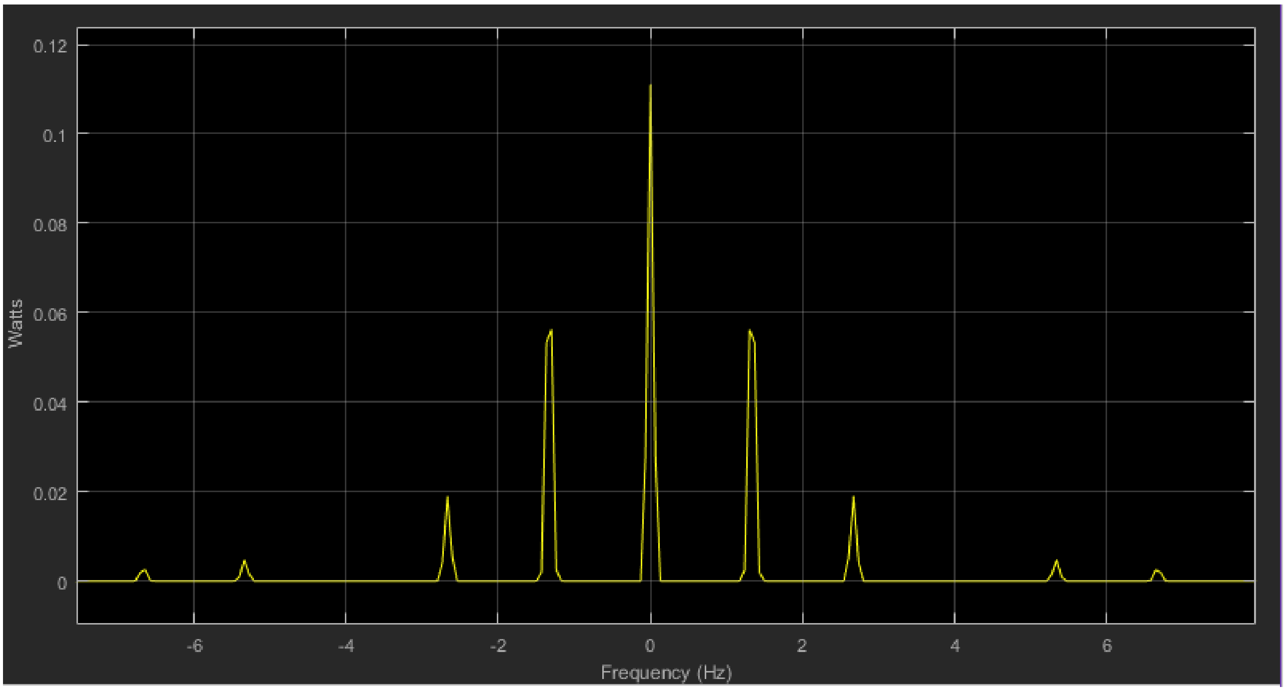


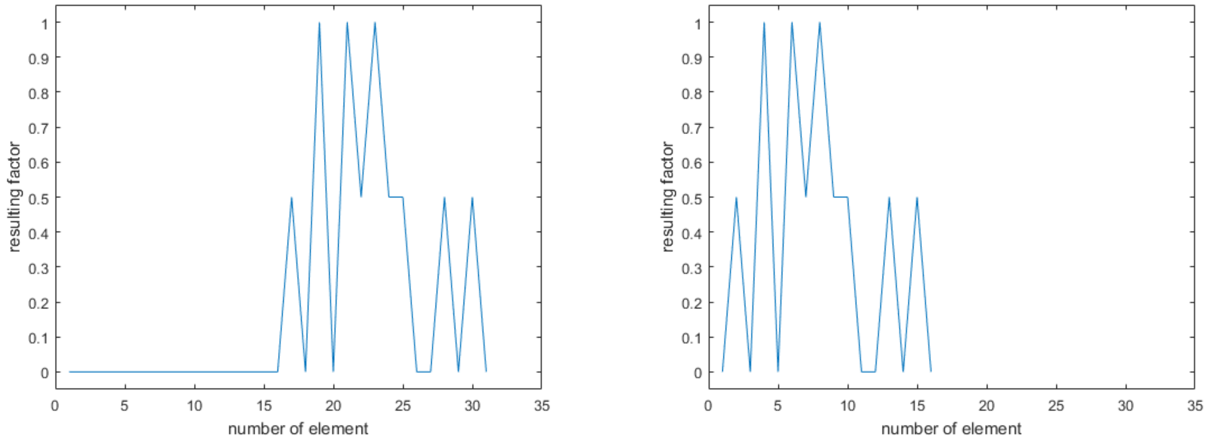
Рис. 4.14: Полученный спектр для дискретного прямоугольного сигнала. Окно Spectrum Analyzer.

Таким образом, были получены спектры различных дискретных прямоугольных сигналов.

5 Корреляция сигналов

В среде Matlab была разработана программа моделирующая передачу 2 байтового сообщения, в котором заключена 1 байтовая посылка.

Для выявления посылки было применено два метода: **ifft** и **xcorr**. Полученные результаты продемонстрированы на рисунке 5.1.

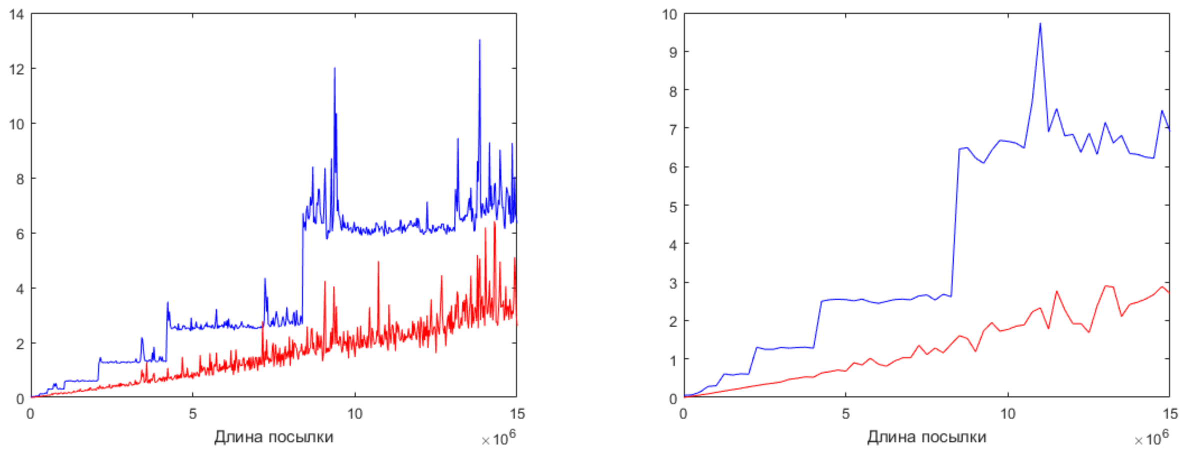


ifft xcorr

Рис. 5.1: Результаты вычисления кросс-корреляции

Была разработана собственная функция **corr**, предназначенная для выявления основной части сообщения. Функция продемонстрированная на листинге 3, была протестирована на различных входных данных листинг 2. Проверка прошла успешно.

Для сравнения времени работы функций **ifft** и **xcorr** использовалась программа с листинга 1. На рисунке 5.2 представлены два графика с различным шагом дискретизации. Красным обозначен – **ifft**, синим – **xcorr**.



с шагом 25 000 с шагом 500 000

Рис. 5.2: Время затрачиваемое на кросс-корреляцию в зависимости от длины посылки

На рисунке 5.2 продемонстрировано преимущества использования **ifft** над **xcorr**.

6 Выводы

Сигналы используются для передачи информации. Сигналы различаются по природе носителя информации, по способу задания сигнала, по непрерывности, по периодичности и по конечности.

Они могут быть представлены как во временной, так и в частотной области. Переход от одного представления к другому можно осуществить с помощью преобразований Фурье. Преобразование применяют потому что при анализе сигналов для одних удобнее временное отображение, а для других частотное.

Корреляционный анализ дает возможность установить в сигналах наличие связи. Методы корреляции активно применяются при анализе случайных процессов для выявления неслучайных составляющих и оценки неслучайных параметров этих процессов.

7 Приложение

Листинг 1: Программа сравнения скорости работы функций

```
close all;
clear all;

X = [0 0 0 1 0 1 0 1 1 1 0 0 0 0 1 0 ];
Y = [1 0 1];

tcorr1 = [];
tcorr2 = [];
tcorr3 = [];
array_with_len    = [];
step_of_for = 500;
len_of_package = step_of_for;

% corr1 = xcorr(X,Y);
% corr1 = corr1 / max(corr1);
% plot(corr1);
% ylim([-0.05,1.05]);
% xlabel('number of element');
% ylabel('resulting factor');
%
% figure;
% F1 = fft(X, length(X));
% F2 = fft(Y, length(X));
% comp = conj(F2);
% corr2 = ifft(comp.*F1/ max(comp)) ;
% plot(corr2);
% xlim([0,35]);
% ylim([-0.05,1.05]);
% xlabel('number of element');
% ylabel('resulting factor');

for i = 0 : 50
    array_with_len = [array_with_len, len_of_package];
   X = randi(2, len_of_package, 1) - 1;
   X = X';

    tic();
    xcorr(Y, X);
    tcorr1 = [tcorr1, toc()];
```

```
    tic();
    F1 = fft(X, length(X));
    F2 = fft(Y, length(X));
    comp = conj(F2);
    corr2 = ifft(comp.*F1);
    tcorr2 = [tcorr2, toc()];

    tic();
    corr(X,Y);
    tcorr3 = [tcorr3, toc()];

    len_of_package = len_of_package + step_of_for
end

plot(array_with_len, tcorr1, 'b', array_with_len, ...
    tcorr2, 'r', array_with_len, tcorr3, 'g');
xlabel('length of package ');
ylabel('execution time, sec');
```

Листинг 2: Программа тестирования функции corr

```
close all;
clear all;

input_signal = [0 0 0 1 0 1 0 1 1 1 0 0 0 0 1 0 ];
synchro_part = [1 0 1];

number_of_true = 0;
for i = (corr(input_signal, synchro_part) ...
        == [0 1 1 1 0 0 0 0])
    number_of_true = number_of_true + i;
end
assert(number_of_true == 8);

%------------------------------------------------------------%

input_signal = [1 1 1 0 1 1 0 0 1 0 1 1 ...
    0 1 1 0 1 0 0 0 0 1 1 0 1];
synchro_part = [0 1 0 1];

number_of_true = 0;
for i = (corr(input_signal, synchro_part) ...
        == [1 0 1 1 0 1 0 0])
    number_of_true = number_of_true + i;
end
assert(number_of_true == 8);

%------------------------------------------------------------%

input_signal = [1 0 1 1 0 1 0 1 1 1 1 1 0 ...
    1 0 0 0 1 0 1 0 1 0 1 0 1 1 0 0 0 1];
synchro_part = [1 1 1 1 1];

number_of_true = 0;
for i = (corr(input_signal, synchro_part) ...
        == [0 1 0 0 0 1 0 1])
    number_of_true = number_of_true + i;
end
assert(number_of_true == 8);
```

Листинг 3: Функция для выявления основной части сообщения

```
function result = corr(input_signal, synchro_part)
% Determines the first occurrence of
%   a parcel in the input signal
%
% Input :    input_signal — input
%            synchro_part — the signal to start sending
%
% Output :   result-the package

for i = 2 : length(synchro_part)
    input_signal = [input_signal 0];
end
number_of_sychro = length(input_signal);
for i = 1 : (length(input_signal) ...
        -length(synchro_part) +1)
    part = input_signal(i : (i +length(synchro_part) -1));
    if (part == synchro_part)
        number_of_sychro = i;
        break;
    end
end
if number_of_sychro < length(input_signal)
    end_of_data = number_of_sychro ...
        +7 +length(synchro_part);
else
    end_of_data = length(input_signal);
end
result = input_signal(number_of_sychro ...
    +length(synchro_part) : end_of_data);
end
```

8 Используемые материалы

1. Correlation (E1.10 Fourier Series and Transforms)
2. Fourier transform (Wikipedia)
3. Signal (Wikipedia)
4. Корреляция сигналов (Давыдов А.В. Теория сигналов и линейных систем)