МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение

высшего образования

«КРЫМСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ им. В. И. ВЕРНАДСКОГО»

ФИЗИКО-ТЕХНИЧЕСКИЙ ИНСТИТУТ

Кафедра компьютерной инженерии и моделирования

**Спектральный анализ непериодических сигналов**

Отчет по лабораторной работе 3

по дисциплине «**Обработка сигналов**»

студента 3 курса группы ИВТ-б-о-202  
Шор Константина Александровича

Направления подготовки 09.03.01«Информатика и вычислительная техника»

Симферополь, 2023

**Лабораторная работа №3**

**Тема:** Спектральный анализ непериодических сигналов

**Цель работы:** Задан одиночный импульс амплитудой Emax и длительностью tи. Определить спектральную плотность импульса. Построить амплитудный и фазовый спектр заданного импульса. Построить спектральные диаграммы. Определить практическую ширину спектра, в котором содержится не менее 95 % энергии одиночного импульса. (Из периодической последовательности импульсов выбирается один импульс, параметры которого приведены в таблице)..

**Теоретические сведения**

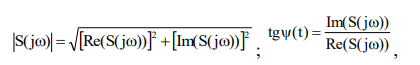
Одиночный импульс может быть представлен как во временной области, так и в частотной. Переход из временной в частотную область осуществляется с помощью прямого преобразования Фурье:

****

Переход из частотной области во временную осуществляется с помощью обратного преобразования Фурье:



Спектральная плотность сигнала представляет собой комплексную величину. Спектральную плотность можно выразить через модуль (амплитудный спектр |S(jω)|) и аргумент (фазовый спектр ψ(t)):



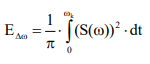
При переходе из временной области в частотную сигнал должен быть абсолютно интегрируемым:



Полная энергия одиночного импульса определяется следующим выражением:



Энергия, сосредоточенная в полосе частот [0÷ωk], определяется, согласно теоремы Парсеваля, следующим соотношением:



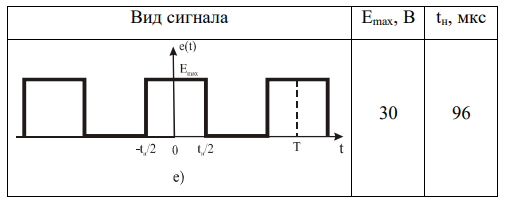
Определение практической ширины спектра (значения ωk) зависит от отношения EΔω/Ec:



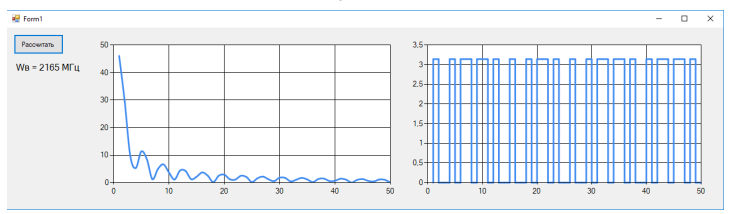
Аналитически рассчитанная функция S(jw):



**Ход работы**



При запуске программы появляется окно, с кнопкой. При нажатии строятся амплитудный и фазовый спектр импульса, и определяется ширина спектра Wв в которой содержится 95% мощности исходного сигнала.



По результатам видно, что ширина спектра в которой содержится 95% энергии одиночного импульса от -2165 до 2165 МГц.

**Вывод:** во время выполнения данной работы был изучен принцип спектрального анализа непериодического сигнала, изучена теоретическая база для понимания работы с комплексными числами. При выполнении данной лабораторной работы были найдены следующие величины: спектральная плотность импульса и практическая ширина спектра. И построены фазовые и амплитудные спектры. Была создана программа, которая вычисляет необходимые величины и строит спектры

**Приложение**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace \_3.Спектральный\_анализ\_непериодических

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

chart2.ChartAreas[0].AxisX.Maximum = 40;

chart1.ChartAreas[0].AxisX.Maximum = 100;

//chart1.ChartAreas[0].AxisY.Maximum = 0.06;

chart1.ChartAreas[0].AxisX.Minimum = 0;

chart2.ChartAreas[0].AxisX.Minimum = 0;

////chart1.ChartAreas[0].AxisY.Minimum = -5;

chart1.Series[0].ChartType = System.Windows.Forms.DataVisualization.Charting.SeriesChartType.Spline;

chart2.Series[0].ChartType = System.Windows.Forms.DataVisualization.Charting.SeriesChartType.Line;

chart1.Series[0].BorderWidth = 3;

chart2.Series[0].BorderWidth = 3;

}

private static double A(double x)

{

return -60 \* Math.Sin(x \* 48) / x; /\*-(60 \* Math.Sin(48 \* x)) / x;\*/

}

private static double Ecm()

{

return 30\*30\*96;

}

public static double Pryam(double a, double b, double n)

{

double S = 0;

double h = (b - a) / n;

for (double x = a; x <= b; x += h)

{

S += h \* A(x);

}

return S/Math.PI;

}

private void button1\_Click(object sender, EventArgs e)

{

double Ew = 0;

int Wv=0;

double Ec=Ecm();

for (int w = 1; Ew < Ec \* 0.95; w++)

{

chart1.Series[0].Points.AddXY(w, Math.Sqrt(Math.Pow(A(w),2)));

Func<double, double> Si = x => Math.Sin(x \* 96 / 2) / x;

Integral B = new Integral(0.000001, w, Si);

double S = 60 / Math.PI \* Math.Abs(B.Simpson(0.000001, w, 10000));

Ew += S;

Wv = w;

}

double[] fh = new double[Wv+1];

for (int i = 0; i < Wv; i++)

{

if(A(i+1) < 0)

{

fh[i] = Math.PI;

}

else

{

fh[i] = 0;

}

}

for (int i = 0; i < Wv; i++)

{

chart2.Series[0].Points.AddXY(i+1, fh[i]);

if (fh[i] > 0)

{

if (fh[i + 1] > 0)

{

}

else

{

chart2.Series[0].Points.AddXY(i + 1, 0);

}

}

else

{

if (fh[i + 1] > 0)

{

chart2.Series[0].Points.AddXY(i + 1, Math.PI);

}

else

{

}

}

}

label1.Text += Wv + " МГц";

}

}

}