МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

ФГБОУ ВО «ВЯТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет автоматики и вычислительной техники

Кафедра радиоэлектронных средств

**Реализация КИХ-фильтра**

Лабораторная работа №1 по дисциплине

"Проектирование устройств и систем на цифровых сигнальных процессорах"

Разработал: студент группы ИНБс-51 \_\_\_\_\_\_\_\_\_\_/Д.С. Шикалов

Руководитель: д.т.н., профессор каф. РЭС \_\_\_\_\_\_\_\_\_\_ /М.А. Земцов

Киров 2022

**Цель работы:** исследование основных типов весовых функций (окон), изучение средств визуализации данных Code Composer Studio.

1. **Код реализации перемножения оконных функций на исходную последовательность**

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

#include "data1.h"

int N = 20;

float Pi=3.141592653589793;

float in\_var1[100]={A\_ARRAY};

float window\_rect[100];

float window\_bartlett[100];

float window\_hann[100];

float in\_var1\_x\_window\_rect[100];

float in\_var1\_x\_window\_bartlett[100];

float in\_var1\_x\_window\_hann[100];

float fft\_in\_var1\_x\_window\_rect[100];

float fft\_in\_var1\_x\_window\_bartlett[100];

float fft\_in\_var1\_x\_window\_hann[100];

void window\_one(){

int i=0;

for(i;i<100;i++){

if(i>=0 && i<=(N-1)){

window\_rect[i]=1;

}

else{

window\_rect[i]=0;

}

}

}

void window\_two(){

int i=0;

for(i;i<100;i++){

if(i<21){

if(i<11){

window\_bartlett[i]=i;

}

else{

window\_bartlett[i]=20-i;

}

}

else{

window\_bartlett[i]=0;

}

}

}

void window\_three(){

int i=0;

for(i;i<100;i++){

if(i<21){

window\_hann[i]=0.5-0.5\*cos(2\*Pi\*i/N);

}

else{

window\_hann[i]=0;

}

}

}

void multiply(){

int i = 0;

for(i=0;i<100;i++){

in\_var1\_x\_window\_rect[i]=in\_var1[i]\*window\_rect[i];

in\_var1\_x\_window\_bartlett[i]=in\_var1[i]\*window\_bartlett[i];

in\_var1\_x\_window\_hann[i]=in\_var1[i]\*window\_hann[i];

}

}

void calculateDFT()

{

float Xr[100];

float Xi[100];

int i, k, n = 0;

for (k = 0; k < 100; k++) {

Xr[k] = 0;

Xi[k] = 0;

for (n = 0; n < 100; n++) {

Xr[k] = (Xr[k] + in\_var1\_x\_window\_rect[n] \* cos(2 \* 3.141592 \* k \* n / 100));

Xi[k] = (Xi[k]- in\_var1\_x\_window\_rect[n] \* sin(2 \* 3.141592 \* k \* n / 100));

}

fft\_in\_var1\_x\_window\_rect[k]=pow(Xr[k]\*Xr[k]+Xi[k]\*Xi[k],0.5);

}

for (k = 0; k < 100; k++) {

Xr[k] = 0;

Xi[k] = 0;

for (n = 0; n < 100; n++) {

Xr[k] = (Xr[k] + in\_var1\_x\_window\_bartlett[n] \* cos(2 \* 3.141592 \* k \* n / 100));

Xi[k] = (Xi[k]- in\_var1\_x\_window\_bartlett[n] \* sin(2 \* 3.141592 \* k \* n / 100));

}

fft\_in\_var1\_x\_window\_bartlett[k]=pow(Xr[k]\*Xr[k]+Xi[k]\*Xi[k],0.5);

}

for (k = 0; k < 100; k++) {

Xr[k] = 0;

Xi[k] = 0;

for (n = 0; n < 100; n++) {

Xr[k] = (Xr[k] + in\_var1\_x\_window\_hann[n] \* cos(2 \* 3.141592 \* k \* n / 100));

Xi[k] = (Xi[k]- in\_var1\_x\_window\_hann[n] \* sin(2 \* 3.141592 \* k \* n / 100));

}

fft\_in\_var1\_x\_window\_hann[k]=pow(Xr[k]\*Xr[k]+Xi[k]\*Xi[k],0.5);

}

}

void main()

{

window\_one();

window\_two();

window\_three();

multiply();

calculateDFT();

}

1. **Графики сигналов во временной области и их спектры**

График входного сигнала во временной области представлен на рисунке 1.

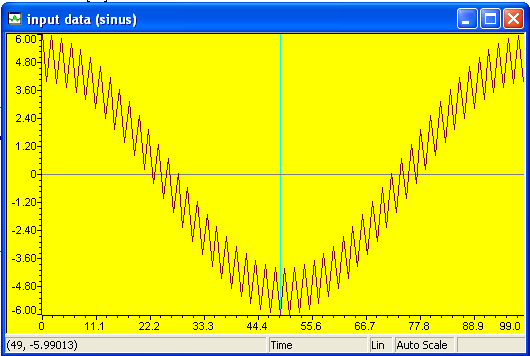


Рисунок 1 – График входного сигнала во временной области

График входного сигнала (прямоугольное окно) во временной области представлен на рисунке 2.

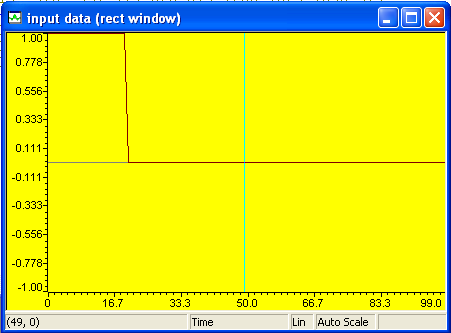


Рисунок 2 – График входного сигнала (прямоугольное окно) во временной области

График входного сигнала (окно Бартлетта) во временной области представлен на рисунке 3.

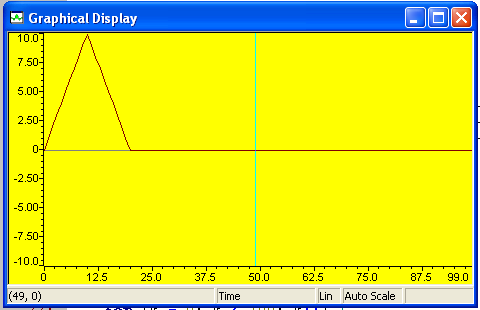


Рисунок 3 – График входного сигнала (окно Бартлетта) во временной области

График входного сигнала (окно Ханна) во временной области представлен на рисунке 4.

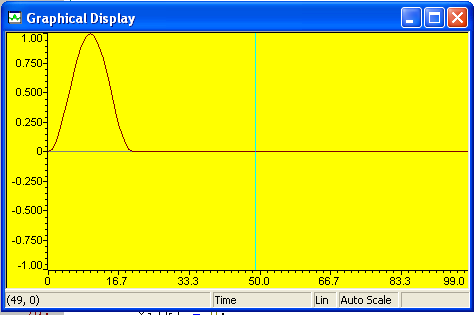


Рисунок 4 – График входного сигнала (окно Ханна) во временной области

График выходного сигнала (перемножение входного сигнала на прямоугольное окно) во временной области представлен на рисунке 5.

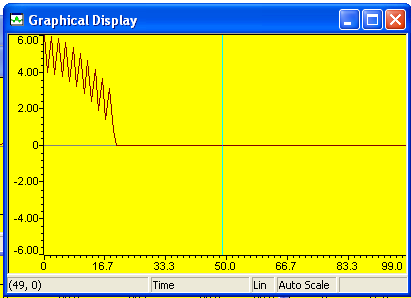


Рисунок 5 – График выходного сигнала (перемножение входного сигнала на прямоугольное окно) во временной области

График выходного сигнала (перемножение входного сигнала на окно Бартлетта) во временной области представлен на рисунке 6.

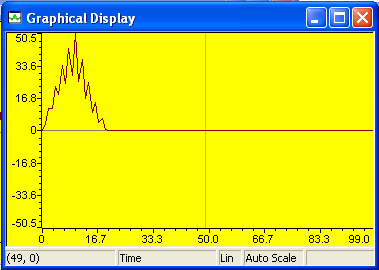


Рисунок 6 – График выходного сигнала (перемножение входного сигнала на окно Бартлетта) во временной области

График выходного сигнала (перемножение входного сигнала на окно Ханна) во временной области представлен на рисунке 7.

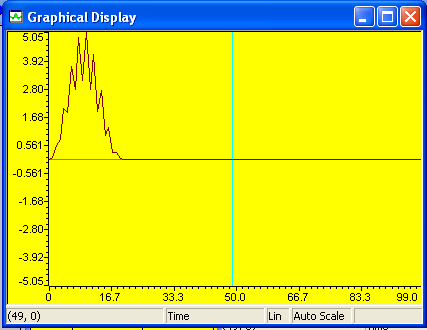


Рисунок 7 – График выходного сигнала (перемножение входного сигнала на окно Ханна) во временной области

Спектр входного сигнала представлен на рисунке 8.

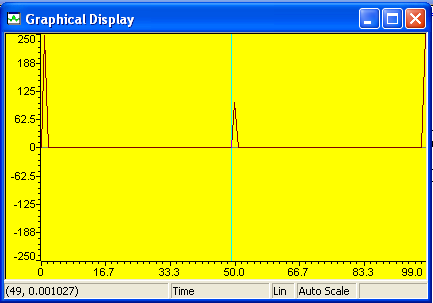


Рисунок 8 – Спектр входного сигнала

Спектр входного сигнала (прямоугольное окно) представлен на рисунке 9.

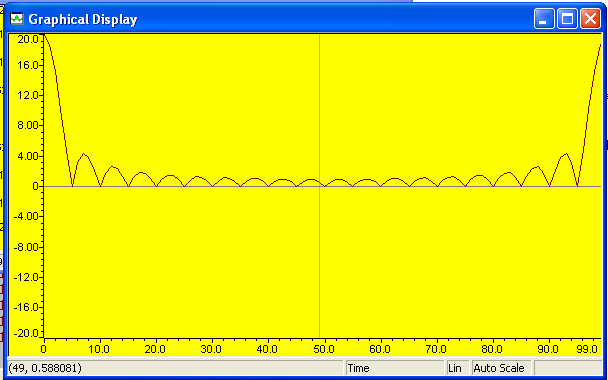


Рисунок 9 – Спектр входного сигнала (прямоугольное окно)

Спектр входного сигнала (окно Бартлетта) представлен на рисунке 10.

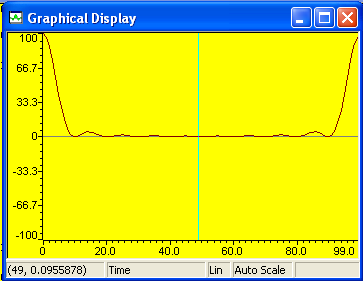


Рисунок 10 – Спектр входного сигнала (окно Бартлетта)

Спектр входного сигнала (окно Ханна) представлен на рисунке 11.

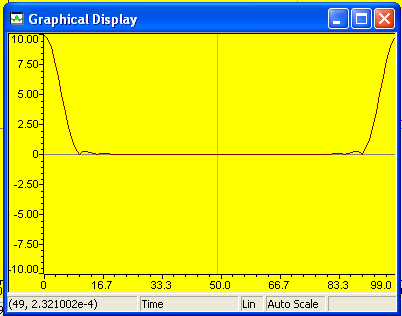


Рисунок 11 – Спектр входного сигнала (окно Ханна)

Спектр выходного сигнала (перемножение входного сигнала на прямоугольное окно) представлен на рисунке 12.

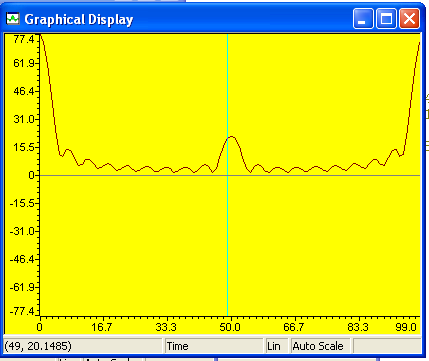


Рисунок 12 – Спектр выходного сигнала (перемножение входного сигнала на прямоугольное окно)

Спектр выходного сигнала (перемножение входного сигнала на окно Бартлетта) представлен на рисунке 13.

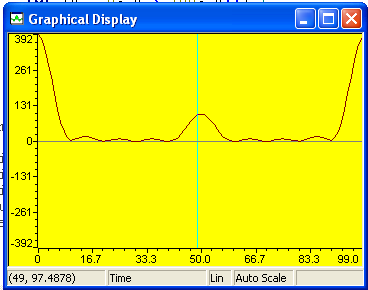


Рисунок 13 – Спектр выходного сигнала (перемножение входного сигнала на окно Бартлетта)

Спектр выходного сигнала (перемножение входного сигнала на окно Ханна) представлен на рисунке 14.

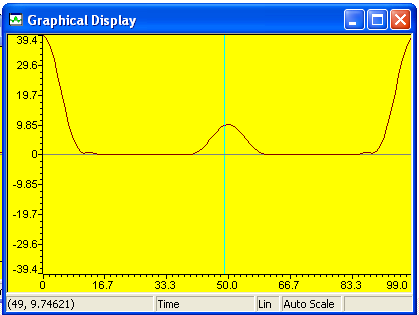


Рисунок 14 – Спектр выходного сигнала (перемножение входного сигнала на окно Ханна)

**Вывод**

Была написана программа на языке программирования “C”в программе Code Composer Studio для платы TMS320C6713 компании Texas Instruments. Код реализует перемножение спектра входного сигнала (по варианту – синусоида с наложенным ВЧ-сигналом) с фильтрующими сигнал окнами (по варианту прямоугольное окно, окно Бартлетта, окно Ханна).

Были получены временные диаграммы входного сигнала, сигнала окон, сигнала перемножения входного сигнала на окно, а также, были получены входного сигнала, сигнала окон, сигнала перемножения входного сигнала на окно.

Спектральные пики на выходе преобразования Фурье расширяются за счет перемножения исходного сигнала на окно.

Боковые лепестки преобразования Фурье изменяют амплитуду соседних спектральных пиков, тем самым, маскируя наличие слабых сигналов.