1. Записать формулу ДПФ и подсчитать количество операций комплексного сложения и комплексного умножения по определению для заданного N.

2. Построить алгоритм Кули-Тьюки для N=N1\* N2. N1-точечное и N2-точечное вычислять в соответствии с заданием.

3. Построить граф Ni-точечного БПФ с прореживанием (в соответствии с заданием для размерности – степени двойки, если таких преобразований 2, то привести граф одного из них). Подсчитать количество операций комплексного сложения и комплексного умножения для каждого алгоритма с длиной – степенью двойки.

4. Подсчитать количество операций комплексного сложения и умножения вычисления N-точечного БПФ по алгоритму Кули-Тьюки.

5. Сравнить количество операций при вычислении N -точечного ДПФ по определению с количеством операций при вычислении N -точечного БПФ по алгоритму Кули-Тьюки.

|  |  |  |
| --- | --- | --- |
| № п/п | Пункт задания | Количество операций |
| 1 |  |  |
| 2 | 1. Отображение входной последовательности в двумерную таблицу в соответствии с (1):   |  |  |  |  | | --- | --- | --- | --- | |  |  |  |  | | n |  |  | | 0 |  |  | | 1 |  |  | | 2 |  |  | | 3 |  |  | | 4 |  |  | | 5 |  |  | | … |  |  | | 959 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | |  |  | |  | |  | |  | |  | |  | |  |  | | 0 | | 1 | | 2 | | ... | | 29 | | 0 | | 0 | | 32 | | 63 | | ... | | 928 | | | 1 | | 1 | | 33 | | 64 | | … | | 929 | | | 2 | | 2 | | 34 | | 65 | | … | | 930 | | | 3 | | 3 | | 35 | | 66 | | … | | 931 | | | … | | … | | … | | … | | … | | … | | | 31 | | 31 | | 62 | | 94 | | … | | 959 | |   2. Вычисление ДПФ длиной точек (внутренняя сумма):  3. N умножений на т.н. поворачивающий множитель :   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | |  |  |  |  |  |  | |  | 0 | 1 | 2 | … | 29 | | 0 |  |  |  |  |  | | 1 |  |  |  | *…* |  | | 2 |  |  |  | *…* |  | | 3 |  |  |  | *…* |  | | … | … | … | … | … | … | | 31 |  |  |  | *…* |  |   4. Вычисление ДПФ длиной точек (внешняя сумма):  5. Отображение таблицы в выходную последовательность в соответствии с (2):   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | |  |  |  |  |  |  | |  | 0 | 1 | 2 | ... | 29 | | 0 | 0 | 1 | 2 | ... | 29 | | 1 | 30 | 31 | 32 | … | 59 | | 2 | 60 | 61 | 62 | … | 89 | | 3 | 90 | 91 | 92 | … | 109 | | … | … | … | … | … | … | | 31 | 930 | 931 | 932 | … | 959 |  |  | | --- | | *k* | | 0 | | 1 | | 2 | | 3 | | 4 | | … | | 959 | | Комплексных  операций нет  N умножений  Комплексных  операций нет |
| 3 |  |  |
| 4 |  |  |
| 5 | Сравнение КТ с алгоритмом по определению |  |

Код программы:

clear;

close all;

clc;

global a;

global m;

a = 0; % Счетчик операций сложения

m = 0; % Счетчик операций умножения

% Параметры сигнала

N = 960; % Длина сигнала

fd = 100; % Частота дискретизации

f1 = 1000; % Частота сигнала

t = 0:N-1; % Временные отсчеты

% Генерация тестового сигнала

X = sin(t \* 2 \* pi \* f1 / fd);

% Отображение исходного сигнала

figure(1);

plot(t, X);

grid on;

title('Исходный сигнал');

xlabel('Отсчеты');

ylabel('Амплитуда');

a=0;

b=0;

% Вычисление ДПФ алгоритмом Кули-Тьюки для N=960

Y1 = kl\_tk960(X);

a

m

% Вычисление ДПФ по определению (прямой метод)

Y2 = opred(X, N);

% Сравнение результатов двух методов

figure(2);

subplot(2,1,1);

plot(t, abs(Y1), '.--g');

grid on;

title('ДПФ по Кули-Тьюки');

xlabel('Отсчеты');

ylabel('Амплитуда');

subplot(2,1,2);

plot(t, abs(Y2), 'r');

grid on;

title('ДПФ по определению');

xlabel('Отсчеты');

ylabel('Амплитуда');

% Обратное преобразование для проверки

X1 = kl\_tk960(Y1') / N;

figure(3);

subplot(2,1,1);

plot(t, X, 'g');

grid on;

title('Исходный сигнал');

xlabel('Отсчеты');

ylabel('Амплитуда');

subplot(2,1,2);

stem(t, real(X1), 'r');

grid on;

title('Восстановленный сигнал (ОДПФ)');

xlabel('Отсчеты');

ylabel('Амплитуда');

% Вывод счетчиков операций

%fprintf('Всего операций сложения: %d\n', a);

%fprintf('Всего операций умножения: %d\n', m);

%% Функция ДПФ по алгоритму Кули-Тьюки для N=960 (32?30)

function [Y] = kl\_tk960(X)

global a;

global m;

N = 960;

N1 = 32; % Размер строки

N2 = 30; % Размер столбца

% 1) Отображение входной последовательности в 2-мерную таблицу 30?32

Y = zeros(N2, N1);

for n = 1:N2

for j = 1:N1

Y(n, j) = X((n-1)\*N1 + j);

end

end

% 2) Вычисление ДПФ каждого столбца (32 ДПФ длиной 30)

for n = 1:N1

temp = opred(Y(:, n), N2);

for j = 1:N2

Y(j, n) = temp(j);

end

end

% 3) Умножение на поворачивающие множители W\_N^(nk)

for n = 1:N2

for k = 1:N1

Y(n, k) = Y(n, k) \* exp(-1i \* 2 \* pi \* (k-1) \* (n-1) / N);

m = m + 1;

end

end

% 4) Вычисление ДПФ каждой строки (30 ДПФ длиной 32)

for n = 1:N2

Y(n, :) = vrem\_32(Y(n, :));

end

% 5) Отображение 2-мерной таблицы в выходную последовательность

Y\_out = zeros(1, N);

for i = 1:N1

for j = 1:N2

Y\_out((i-1)\*N2 + j) = Y(j, i);

end

end

Y = Y\_out;

end

%% Функция вычисления ДПФ по определению

function Y = opred(X, N)

global a;

global m;

Y = zeros(1, N);

for k = 1:N

for n = 1:N

Y(k) = Y(k) + X(n) \* exp(-1i \* 2 \* pi \* (n-1) \* (k-1) / N);

a = a + 1;

m = m + 1;

end

end

end

%% Оптимизированная функция вычисления ДПФ для N=32 (используется в Кули-Тьюки)

function Y = vrem\_32(A)

global a;

global m;

N = 32;

% 1) Этап перестановки (бит-реверс)

NV2 = round(N/2);

NM1 = N-1;

J = 1;

for I = 1:NM1

if I < J

% Меняем местами элементы

T = A(J);

A(J) = A(I);

A(I) = T;

end

K = NV2;

while K < J

J = J - K;

K = round(K/2);

end

J = J + K;

end

% 2) Предварительное вычисление поворачивающих множителей

koef = zeros(1, 512);

for index = 1:512

tx = pi/512 \* (index-1);

koef(index) = complex(cos(tx), -sin(tx));

end

% 3) Основной цикл БПФ (5 этапов для N=32)

DIND = round(N/2);

LE = 1;

for L = 1:5

LE1 = LE;

LE = LE \* 2;

IND = 1;

for J = 1:LE1

I = J;

INDM = round((IND-1)\*1024/N + 1);

while I <= N

IP = I + LE1;

T = A(IP) \* koef(INDM);

m = m + 1;

A(IP) = A(I) - T;

A(I) = A(I) + T;

a = a + 2;

I = I + LE;

end

IND = IND + DIND;

end

DIND = round(DIND/2);

end

Y = A;

end

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Алгоритм | Требование к N | Количество сложений | Количество умножений | Выигрыш |
| Дпф | Любое N | A = N\*(N-1) | M = N\*N | 0, основная мера |
| Кули-Тьюки | N = N1\*N2 | A = N1 \* A(N2) + N2 \*A(N1) | M = N1 \* M(N2) + N2 \* M(N1) + N | A = = =  M = = = |
| Бпф | N=2k | A = N\*log2N | M = N/2\*log2N | A =  M = |