МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ

Ордена Трудового Красного Знамени федеральное государственное бюджетное образовательное учреждение высшего образования

МОСКОВСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ СВЯЗИ И ИНФОРМАТИКИ

Кафедра Многоканальные телекоммуникационные системы

Курсовая работа

Выполнил: студент группы

БПАС2351 Садыкова П.И.

Проверил: Шаврин С.С.

Москва 202

1. Задание на курсовую работу

На рисунке 1 показано задание на проектирование КИХ-фильтра 80 порядка:

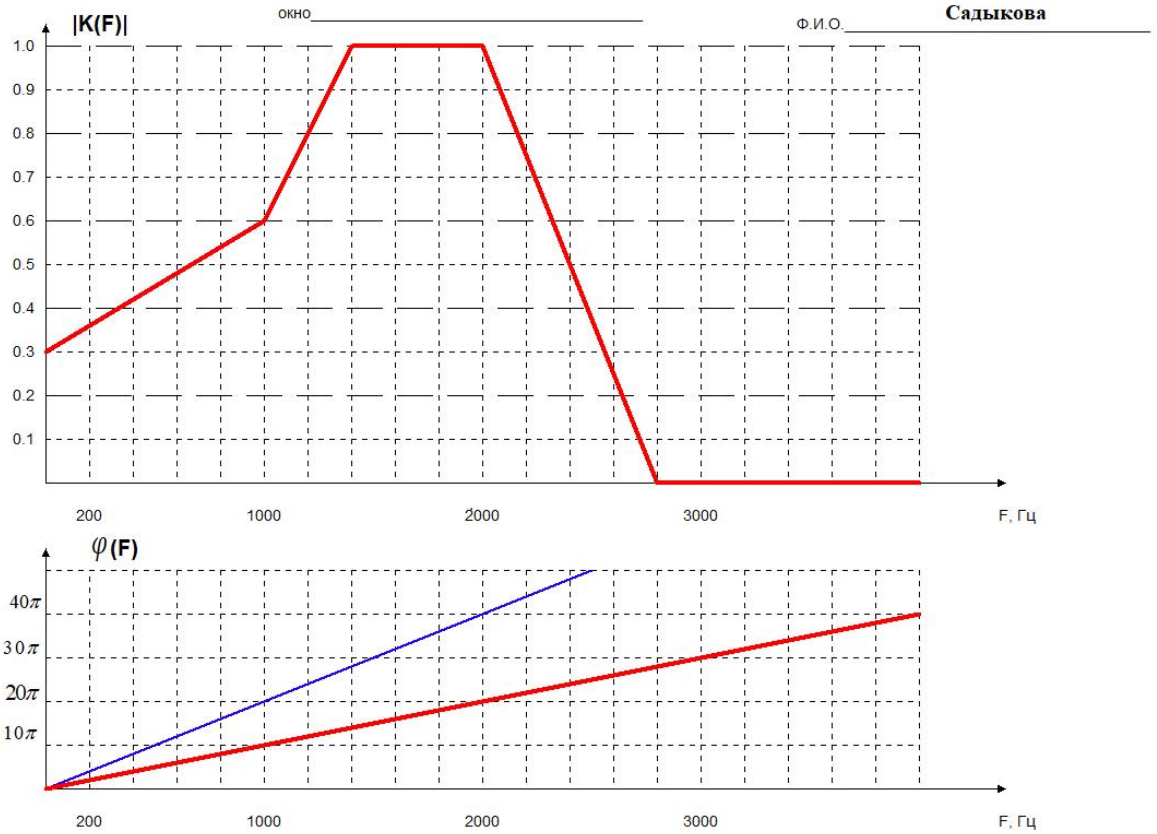


Рисунок 1 - Исходные параметры АЧХ и ФЧХ проектируемого КИХ-фильтра.

В ходе работы необходимо:

1) аппроксимировать заданные АЧХ и ФЧХ отрезками прямых и получить их уравнения;

2) вычислить отсчеты комплексного коэффициента передачи с шагом 100 Гц, достроить его до полного периода частоты дискретизации и обратным дискретным преобразованием Фурье получить импульсную характеристику;

3) перевести импульсную характеристику в 16-разрядные коэффициенты фильтра в пакете SciLab;

2 ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

2.1 Архитектура процессора ADSP-2181

ADSP-2181 - 16-разрядный сигнальный процессор с фиксированной точкой. Раздельная память программ и данных позволяет за один командный цикл выбирать команду и два операнда. Структурная схема процессора приведена на рисунке 2.

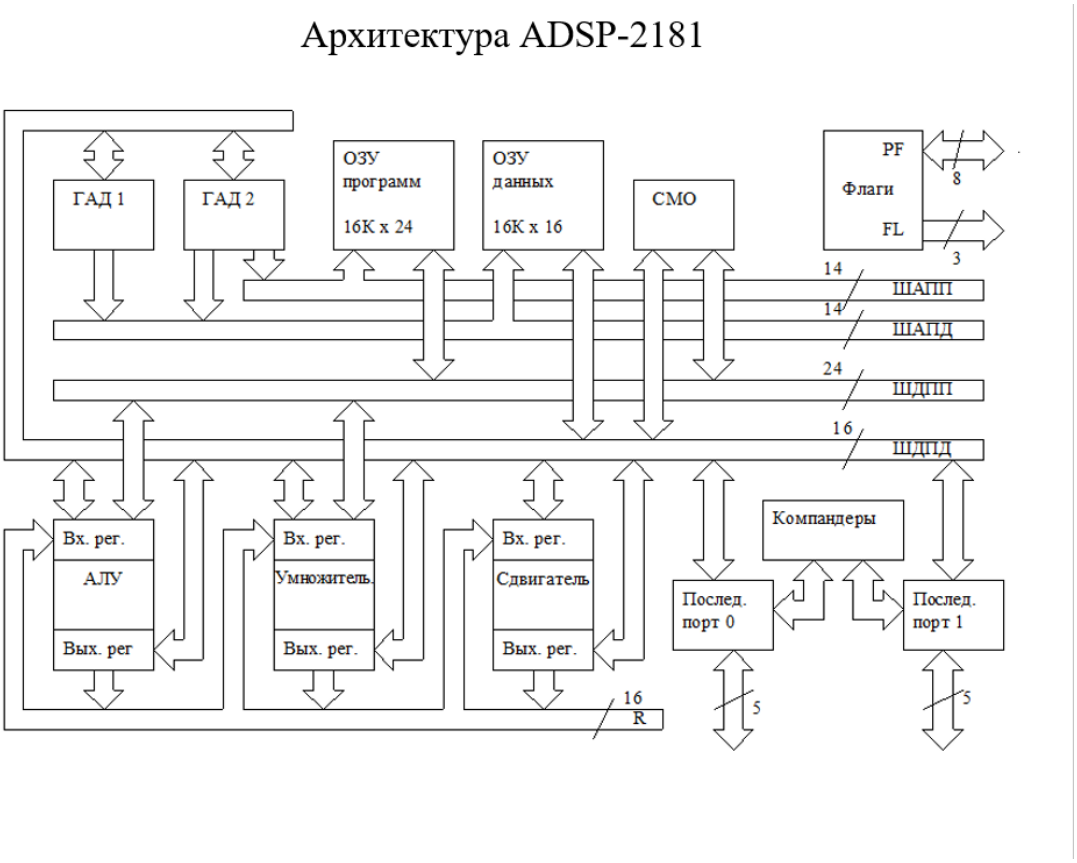


Рисунок 2 - Структурная схема процессора ADSP-2181

Основные узлы процессора:

- арифметико-логическое устройство для арифметических и логических операций над 16-разрядными операндами;

- умножитель-накопитель для умножения двух 16-разрядных чисел и накопления результата в 40-разрядном регистре MR;

- сдвигатель для сдвига, нормализации и определения порядка числа;

- генераторы адресов ГАД1 и ГАД2 для обращения к памяти и организации кольцевых буферов;

- память программ на 16К 24-разрядных слов и память данных на 16К 16-разрядных слов;

- последовательные порты SPORT0 и SPORT1, таймер и порты прямого доступа к памяти.

Обмен между узлами выполняется по внутренним шинам адреса, данных и результата. Процессор может совмещать умножение с накоплением и чтение операндов, что используется при вычислении выходных отсчетов фильтра.

2.2 Арифметико-логическое устройство

АЛУ выполняет операции над двумя 16-разрядными операндами. Операнд X поступает из регистров AX0, AX1 или с шины результата, операнд Y - из AY0, AY1 или AF. Результат записывается в AR или AF. Для обработки прерываний предусмотрен дополнительный комплект регистров. Структура АЛУ и формат команд показаны на рисунке 3.

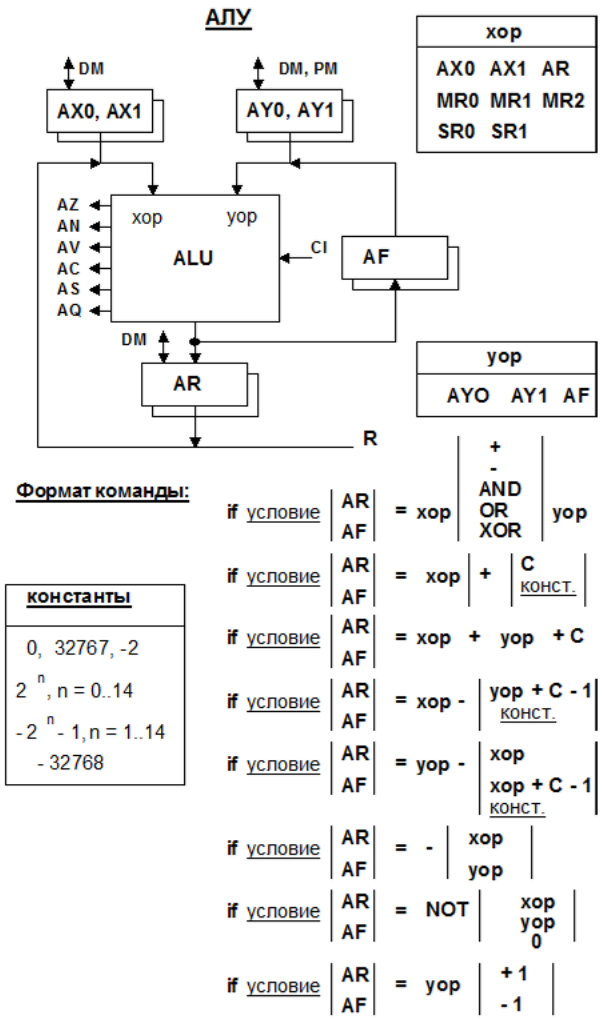


Рисунок 3 - Структура и формат команд АЛУ

После выполнения операций устанавливаются признаки результата: AZ - нуль, AN - отрицательный результат, AV - переполнение, AC - перенос. Признаки AS и AQ используются при вычислении модуля и делении. Условные команды выполняются в зависимости от состояния этих признаков. Режим насыщения включается командой ena ar_sat.

2.3 Умножитель-накопитель

Умножитель-накопитель перемножает 16-разрядные операнды и складывает результат с содержимым 40-разрядного регистра MR либо вычитает его. Поддерживаются знаковый, беззнаковый и смешанный форматы операндов. Структура узла приведена на рисунке 4.

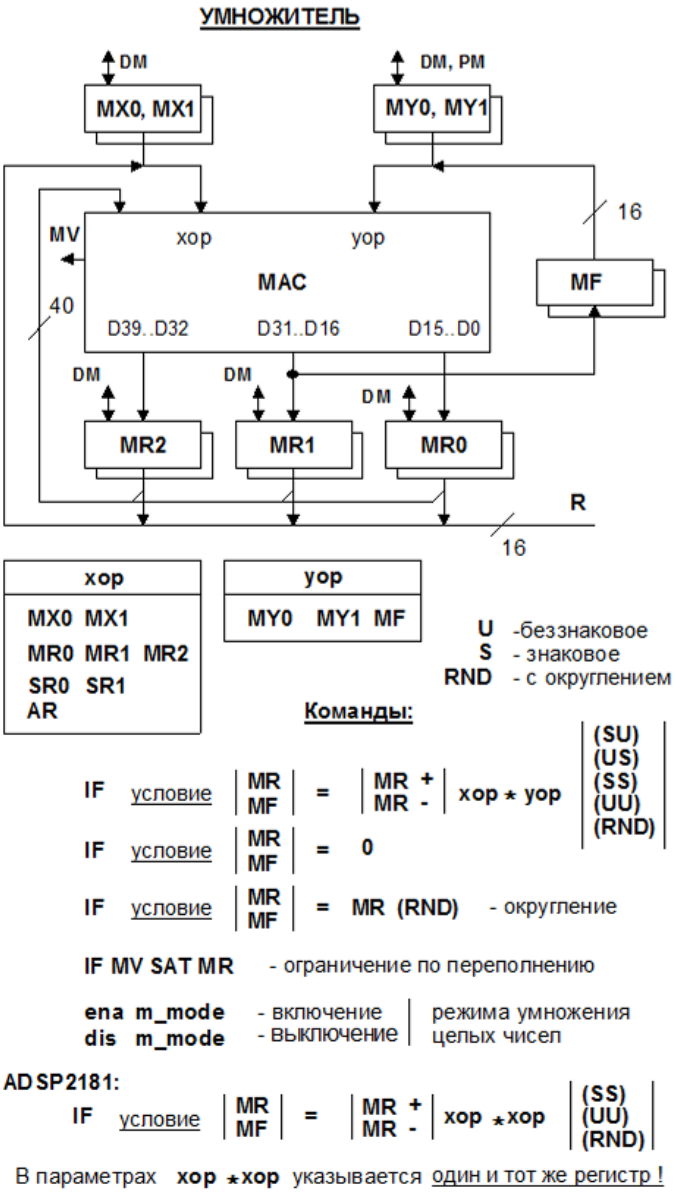


Рисунок 4 - Структура и формат команд умножителя-накопителя

Защитные разряды MR2 используются при накоплении суммы произведений. В программе фильтра команда mr = mr (rnd) округляет результат, а if mv sat mr ограничивает его при переполнении.

2.4 Сдвигатель

Сдвигатель выполняет арифметические и логические сдвиги в 32-разрядном поле SR1:SR0, нормализацию и определение порядка числа. Величина сдвига задается константой или регистром SE. Структура узла показана на рисунке 5.

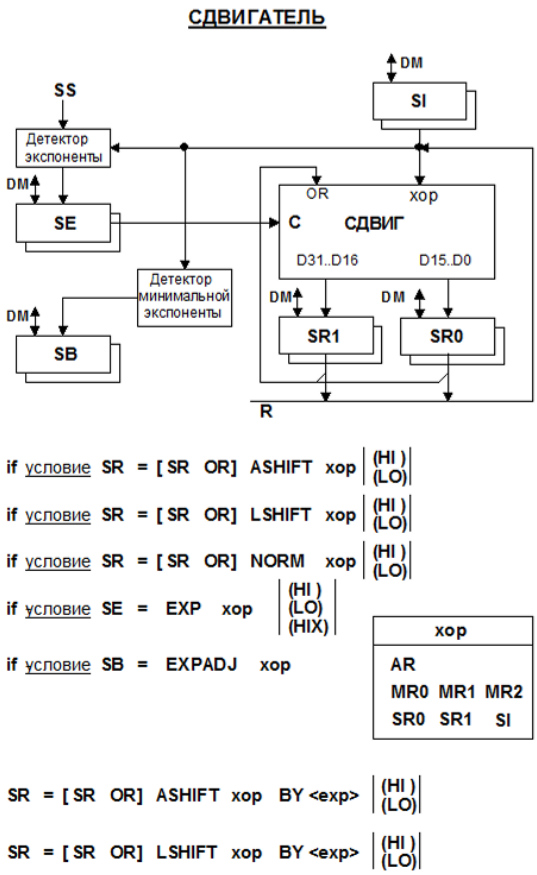


Рисунок 5 - Структура и формат команд сдвигателя

2.5 Представление чисел в процессоре

В процессоре используются 16-разрядные целые и дробные числа со знаком. Целые числа представлены в диапазоне от -32768 до 32767, коэффициенты фильтра - в дробном формате 1.15. Для записи в память значения импульсной характеристики умножаются на 32768 и округляются до целого. Форматы представления чисел показаны на рисунке 6.



Рисунок 6 - Диапазоны представления чисел в процессоре

2.6 КИХ-фильтр и метод частотной выборки

Каждый выходной отсчет КИХ-фильтра вычисляется как взвешенная сумма входных отсчетов:

$$y(n) = \sum_{k=0}^{N-1} h\,(k)\,x(n-k), \qquad (1)$$

Здесь x(n) и y(n) - входной и выходной сигналы, h(k) - коэффициенты фильтра. В расчете используется 80 коэффициентов.

Комплексный коэффициент передачи фильтра связан с импульсной характеристикой дискретным преобразованием Фурье:

$$H(F) = \sum_{n=0}^{N-1} h\,(n)\,e^{-j2\pi Fn/F_д} = K(F)\,e^{j\varphi(F)}. \qquad (2)$$

При частотной выборке характеристика задается в 80 равноотстоящих точках с шагом 100 Гц. Импульсная характеристика определяется обратным дискретным преобразованием Фурье:

$$h(n) = \frac{1}{N}\sum_{k=0}^{N-1} H\,(k)\,e^{j2\pi kn/N}, \quad n = 0,1,\dots,N-1. \qquad (3)$$

Чтобы h(n) получилась вещественной, отсчеты H(k) на интервале $F_д/2...F_д$ должны быть комплексно сопряжены с отсчетами на интервале $0...F_д/2$:

$$H(N-k) = H^*(k), \quad k = 1,2,\dots,N/2-1. \qquad (4)$$

В узлах частотной сетки фильтр воспроизводит заданную характеристику. Между узлами возможны отклонения и пульсации, которые оцениваются при проверке фильтра.

3. Расчёт КИХ-фильтра

Для начала следует аппроксимировать АЧХ и ФЧХ. В заданном варианте АЧХ есть 7 отрезков. Необходимо найти уравнения данных прямых:

1. От 0 до 1000 Гц:

Уравнение прямой: $y = kx + b$

$(0; 0{,}3), (1000; 0{,}6)$

$$\begin{cases} 0{,}3 = 0k + b \\ 0{,}6 = 1000k + b \end{cases} => \begin{cases} b = 0{,}3 \\ k = \dfrac{0{,}6 - 0{,}3}{1000} \end{cases} => \begin{cases} b = 0{,}3 \\ k = 0{,}0003 \end{cases} =>$$

$y = 0{,}0003x + 0{,}3$

2. От 1000 до 1400 Гц:

(1000;0,6) и (1400;1,0)

$$\begin{cases} 0{,}6 = 1000k + b \\ 1.0 = 1400k + b \end{cases} => \begin{cases} b = 0{,}6 - 1 \\ k = \dfrac{0{,}4}{400} \end{cases} => y = 0{,}001x - 0{,}4$$

3. От 1400 до 2000 Гц:

$(1400; 1.0), (2000; 1.0)$

Поскольку отрезок параллелен $y = 1.0$

4. От 2000 до 2800 Гц:

$$(2000; 1.0), (2800; 0) => \begin{cases} 1.0 = 2800k + b \\ 0 = 2000k + b \end{cases} =>$$

$$=> \begin{cases} b = -1200k \\ 1.0 = 2000k - 2800k \end{cases} => \begin{cases} b = 3.5 \\ k = -0{,}00125 \end{cases} => y = -0{,}00125x + 3{,}5$$

5. От 2800 Гц:

$(2800; 0), (4000; 0)$

Поскольку, отрезок лежит на оси $y = 0$

В заданном ФЧХ необходимо найти уравнения данной прямой:

Уравнение прямой фазы также имеет вид y = kF + b. Фаза выходит из нуля, поэтому b = 0

1. Проходит через точки (0; 0) и (2000; 20π):

$$=> \begin{cases} b = 0 \\ 20\pi = 2000k \end{cases} => 0.01\pi$$

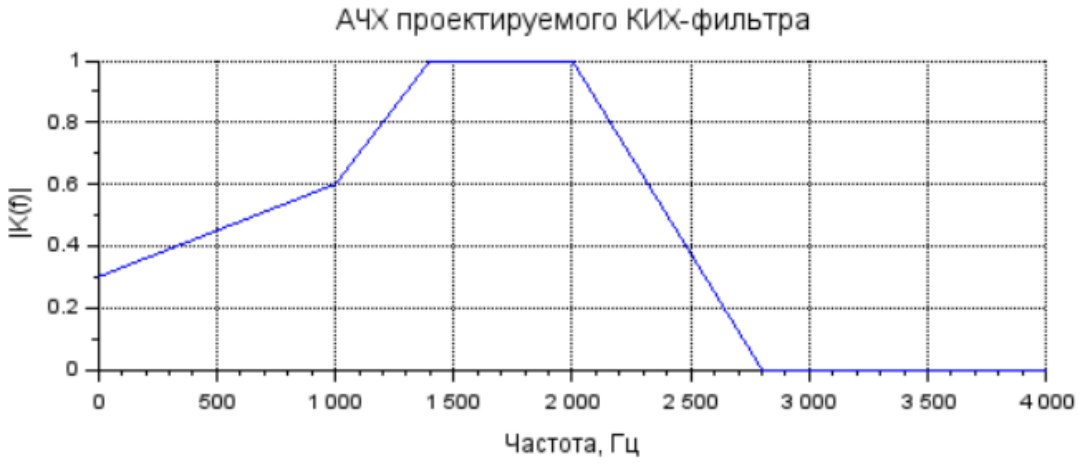
Для удобства внесем данные аппроксимации ФЧХ и ФЧХ в Таблице № 1 с шагом дискретизации Δf = 100Гц. (от 0 до 4000 Гц)

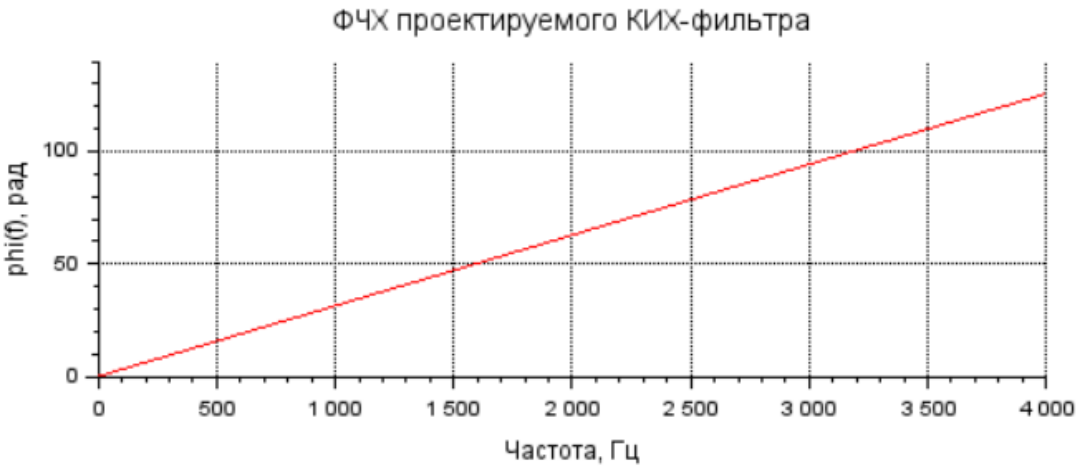
Таблица 1 - Аппроксимация АЧХ и ФЧХ

n	f	K(f)	φ(f)
0	0	0,300	0,000
1	100	0,330	3,142
2	200	0,360	6,283
3	300	0,390	9,425
4	400	0,420	12,566
5	500	0,450	15,708
6	600	0,480	18,850
7	700	0,510	21,991
8	800	0,540	25,133
9	900	0,570	28,274
10	1000	0,600	31,416
11	1100	0,700	34,558
12	1200	0,800	37,699
13	1300	0,900	40,841
14	1400	1,000	43,982
15	1500	1,000	47,124
16	1600	1,000	50,265
17	1700	1,000	53,407
18	1800	1,000	56,549
19	1900	1,000	59,690
20	2000	1,000	62,832
21	2100	0,875	65,973
22	2200	0,750	69,115
23	2300	0,625	72,257
24	2400	0,500	75,398
25	2500	0,375	78,540
26	2600	0,250	81,681
27	2700	0,125	84,823
28	2800	0,000	87,965

29	2900	0,000	91,106
30	3000	0,000	94,248
31	3100	0,000	97,389
32	3200	0,000	100,531
33	3300	0,000	103,673
34	3400	0,000	106,814
35	3500	0,000	109,956
36	3600	0,000	113,097
37	3700	0,000	116,239
38	3800	0,000	119,381
39	3900	0,000	122,522
40	4000	0,000	125,664

Графики АЧХ и ФЧХ представлены на рисунках 2,3:





Теперь необходимо построить график действительной и мнимой части от АЧХ и ФЧХ, использовав формулу:

$$\begin{cases} Re[K(n)] = |K(n)| * cos\,[\varphi(n)] \\ Im[K(n)] = |K(n)| * sin[\varphi(n)] \end{cases}, 0 \leq \text{n} \leq 40$$

В таблице 2 представлены значения действительной и мнимой частей.

Таблица 2 - Значения мнимой и действительной частей АЧХ и ФЧХ

n	f	$Re[K(\mathrm{f})]$	$Im[K(\mathrm{f})]$
0	0	0,3000	**0,0000**
1	100	-0,3300	**0,0000**
2	200	0,3600	**0,0000**
3	300	-0,3900	**0,0000**
4	400	0,4200	**0,0000**
5	500	-0,4500	**0,0000**
6	600	0,4800	**0,0000**
7	700	-0,5100	**0,0000**
8	800	0,5400	**0,0000**
9	900	-0,5700	**0,0000**
10	1000	0,6000	**0,0000**
11	1100	-0,7000	**0,0000**
12	1200	0,8000	**0,0000**
13	1300	-0,9000	**0,0000**
14	1400	1,0000	**0,0000**
15	1500	-1,0000	**0,0000**
16	1600	1,0000	**0,0000**
17	1700	-1,0000	**0,0000**
18	1800	1,0000	**0,0000**
19	1900	-1,0000	**0,0000**
20	2000	1,0000	**0,0000**
21	2100	-0,8750	0,0000
22	2200	0,7500	0,0000
23	2300	-0,6250	0,0000
24	2400	0,5000	0,0000
25	2500	-0,3750	0,0000
26	2600	0,2500	0,0000
27	2700	-0,1250	0,0000
28	2800	0,0000	0,0000

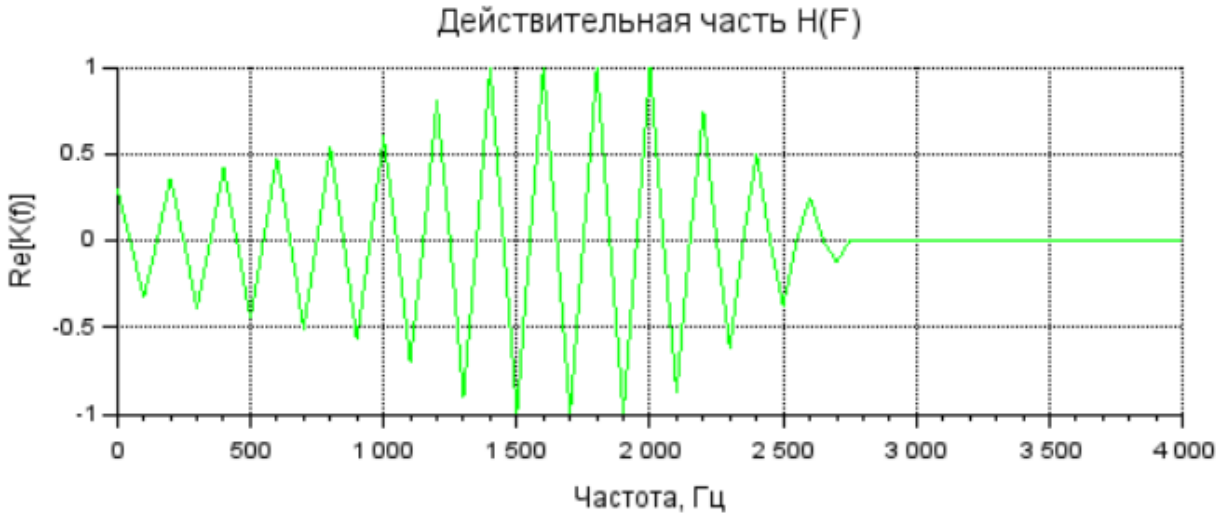
29	2900	0,0000	0,0000
30	3000	0,0000	0,0000
31	3100	0,0000	0,0000
32	3200	0,0000	0,0000
33	3300	0,0000	0,0000
34	3400	0,0000	0,0000
35	3500	0,0000	0,0000
36	3600	0,0000	0,0000
37	3700	0,0000	0,0000
38	3800	0,0000	0,0000
39	3900	0,0000	0,0000
40	4000	0,0000	0,0000

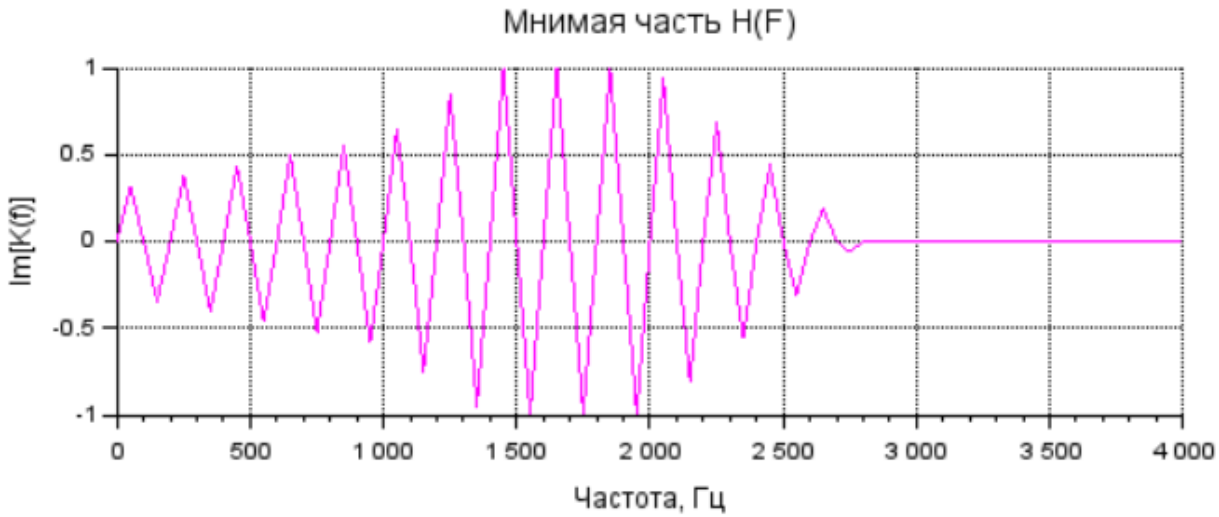
Далее строится график реальной части комплексного коэффициента передачи по полученным ранее точкам, а также график мнимой части КЧХ в диапазоне от 0 до $F_д$ /2 (4000 Гц):

Рисуноки 4,5 - Действительная часть КЧХ в диапазоне $0 \leq f < F$д/2, мнимая часть КЧХ в диапазоне $0 \leq f < F$д/2.

Функция достраивается согласно правилу:

$$a^*(f) = \begin{cases} a(f), & \text{если } f \leq 4000 \\ a(4000 - f), & \text{если } f > 4000 \end{cases}$$

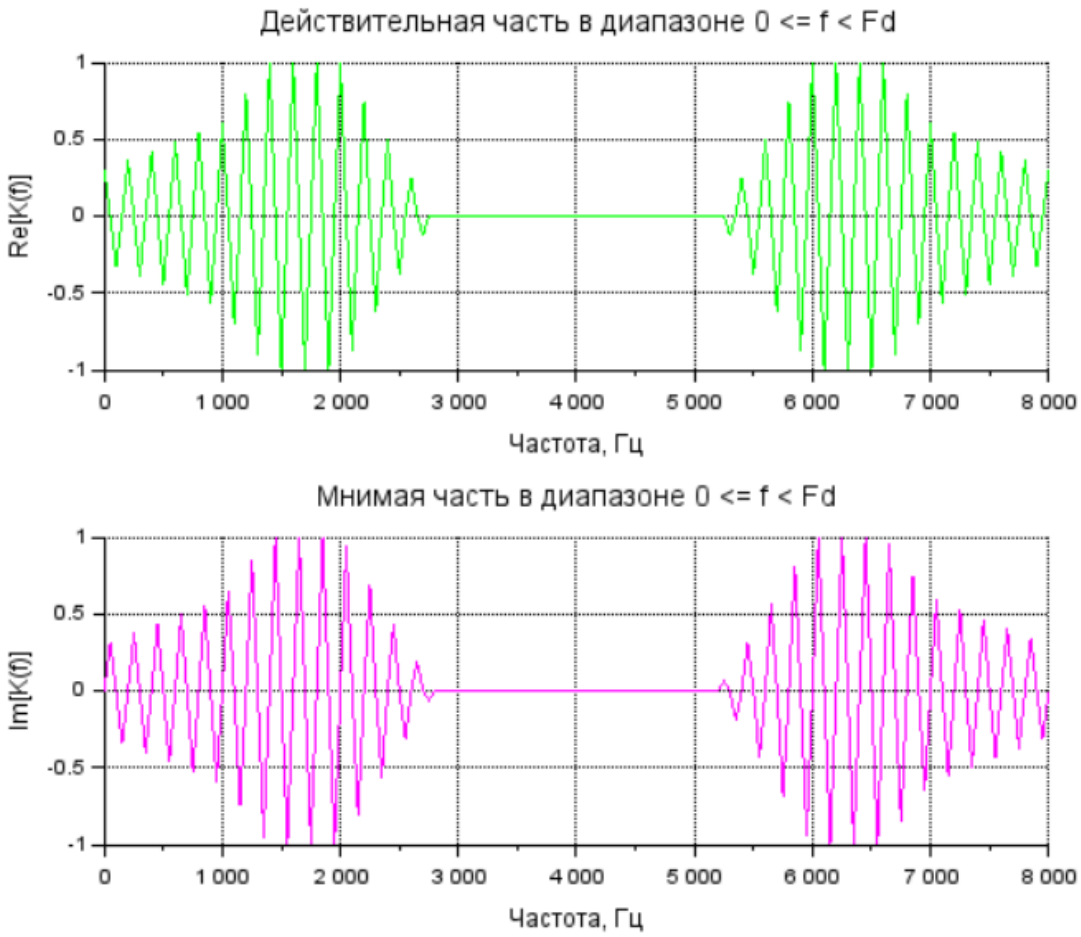




$$b^*(f) = \begin{cases} b(f), & \text{если } f \leq 4000 \\ -b(4000 - f), & \text{если } f > 4000 \end{cases}$$

где $f \epsilon [0 \ldots 4000]$

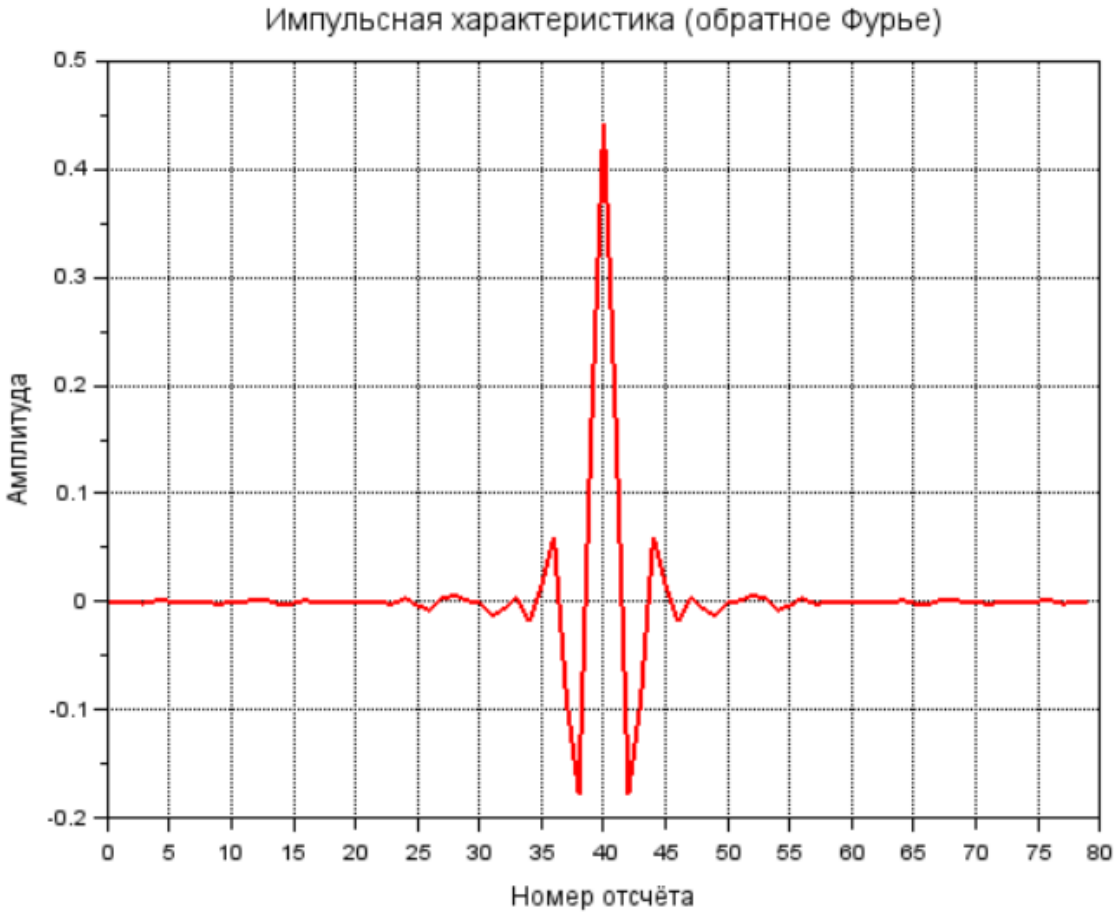
Рисунок 2 и 7 - Действительная часть в диапазоне $0 \leq f < F$д, мнимая часть в диапазоне $0 \leq f < F$д



Необходимо получить импульсную характеристику через обратное преобразование Фурье:

$$h(t) = F^{-1}\left(H^*(f)\right) = F^{-1}\left(a^*(f) + jb^*(f)\right) = \int_{-\infty}^{+\infty} H^*(f)e^{i2\pi\omega t}d\omega$$

Рисунок 8 – Импульсная характеристика



Для дальнейшего выполнения задания нам необходимо получить коэффициенты фильтра, для это нужно выполнить обратное преобразование Фурье в программе Scilab. Ниже представлены полученные значения.

164, -57, 156, -51, 148, -43, 141, -34, 134, -23, 126, -10, 118, 5, 108, 23, 97, 45, 83, 70, 65, 101, 41, 138, 8, 185, -39, 245, -106, 324, -206, 432, -361, 586, -616, 823, -1080, 1233, -2142, 2112, 14500, 2112, -2142, 1233, -1080, 823, -616, 586, -361, 432, -206, 324, -106, 245, -39, 185, 8, 138, 41, 101, 65, 70, 83, 45, 97, 23, 108, 5, 118, -10, 126, -23, 134, -34, 141, -43, 148, -51, 156, -57;

4 РЕАЛИЗАЦИЯ РАСЧЕТА В SCILAB

В SciLab заданы узловые точки АЧХ и ФЧХ. Программа выполняет интерполяцию с шагом 100 Гц, достраивает спектр до 80 отсчетов, вычисляет обратное БПФ и записывает коэффициенты в файл re.dat. Ниже приведен текст программы.

```
    points = [  0,    0.3;
              1000, 0.6;
              1400, 1.0;
              2000, 1.0;
              2800, 0.0;
             4000, 0.0];

    // Линейная интерполяция
    freq_points = points(:, 1);
    amp_points  = points(:, 2);
    num_points = size(points, 1);

    // Обработка границ
    if F < freq_points(1) then ach = amp_points(1); return;
    elseif F > freq_points(num_points) then ach = amp_points(num_points); return;
    end

    // Линейная интерполяция (нахождение отрезка)
    for i = 1:(num_points - 1)
       x1 = freq_points(i); x2 = freq_points(i+1);
       if F >= x1 & F <= x2 then
          y1 = amp_points(i); y2 = amp_points(i+1);
          if (x2 - x1) == 0 then ach = y1; return; end
          slope = (y2 - y1) / (x2 - x1);
          ach = y1 + (F - x1) * slope;
          return;
       end
    end
endfunction

// Функция для Фазо-Частотной Характеристики (ФЧХ) по красной линии
function fch = fi(F)
    // УЗЛОВЫЕ ТОЧКИ ФЧХ (красная линия): [Частота (Гц), Фаза (радианы)]
    points = [  0,    0;
              2000, 20 * %pi;
              4000, 40 * %pi];

    // Линейная интерполяция
    freq_points = points(:, 1);
    phase_vals_rad = points(:, 2);
    num_points = size(points, 1);
```

```
    // Обработка границ
    if F < freq_points(1) then fch = phase_vals_rad(1); return;
    elseif F > freq_points(num_points) then fch = phase_vals_rad(num_points); return;
    end

    // Линейная интерполяция (нахождение отрезка)
    for i = 1:(num_points - 1)
        x1 = freq_points(i); x2 = freq_points(i+1);
        if F >= x1 & F <= x2 then
            y1 = phase_vals_rad(i); y2 = phase_vals_rad(i+1);
            if (x2 - x1) == 0 then fch = y1; return; end
            slope = (y2 - y1) / (x2 - x1);
            fch = y1 + (F - x1) * slope;
            return;
        end
    end
endfunction

// Расчет АЧХ/ФЧХ, построение графиков и заполнение таблицы аппроксимации
ACH = zeros(1, 41);
FCH = zeros(1, 41);
F = 0 : 100 : 4000;
table_ach_fch = zeros(41, 4);
for i = 1:41
    ACH(i) = K(F(i));
    FCH(i) = fi(F(i));
    table_ach_fch(i, :) = [i-1, F(i), ACH(i), FCH(i)];
end

scf(0); clf();
xgrid();
xtitle('ACH', 'F, Hz', '|K(F)|');
plot(F, ACH);
a = gca();
a.data_bounds = [min(F)-0.5, min(ACH)-0.1; max(F)+0.5, max(ACH)+0.1];

scf(1); clf();
xgrid();
xtitle('FCH', 'F, Hz', 'fi(F)');
plot(F, FCH, 'r');
a = gca();
a.data_bounds = [min(F)-0.5, min(FCH)-1; max(F)+0.5, max(FCH)+5];

// Нахождение действительной и мнимой части ДО расширения АЧХ и ФЧХ
table_re_im = zeros(41, 4);
Re_H_half = zeros(1, 41);
Im_H_half = zeros(1, 41);
for i = 1:41
    Re_H_half(i) = abs(ACH(i)) * cos(FCH(i));
    Im_H_half(i) = abs(ACH(i)) * sin(FCH(i));
    table_re_im(i, :) = [i-1, F(i), Re_H_half(i), Im_H_half(i)];
```

```
end
headers_re_im = ["N", "F", "Re", "Im"];
csvWrite([headers_re_im; string(table_re_im)], "re_im.csv", ";");
disp("Таблица с заголовками сохранена в re_im.csv");
disp(table_re_im);

scf(2); clf();
xtitle('Re[H(F)]', 'F, Hz', 'a*(F)');
xgrid();
plot(F, Re_H_half, 'r');

scf(3); clf();
xtitle('Im[H(F)]', 'F, Hz', 'b*(F)');
xgrid();
plot(F, Im_H_half, 'b');

// Нахождение действительной и мнимой части ПОСЛЕ расширения АЧХ и ФЧХ
Re = zeros(1, 80);
Im = zeros(1, 80);
Hfunc = zeros(1, 80);
F_extended = 0 : 100 : 7900;

for i = 1:41
   Re(i) = abs(ACH(i)) * cos(FCH(i));
   Im(i) = abs(ACH(i)) * sin(FCH(i));
   Hfunc(i) = Re(i) + %i * Im(i);
   if i >= 2 & i <= 40 then
      Hfunc(82-i) = Re(i) - %i * Im(i);
      Re(82-i) = Re(i);
      Im(82-i) = -Im(i);
   end
end

scf(4); clf();
xgrid();
xtitle('Re[H(F)]', 'F, Hz', 'a*(F)');
plot(F_extended, Re, 'r');
a = gca();
a.data_bounds = [min(F_extended)-0.5, min(Re)-0.1; max(F_extended)+0.5, max(Re)+0.1];

scf(5); clf();
xgrid();
xtitle('Im[H(F)]', 'F, Hz', 'b*(F)');
plot(F_extended, Im, 'b');

// Обратное преобразование Фурье
IX = ifft(Hfunc);

scf(6); clf();
xgrid();
xtitle('IX', 'i [отсчеты]', 'h(t)');
plot(0:79, imag(IX), 'b');
```

```
plot(0:79, real(IX), 'r');

// Вывод в файл
rex = real(IX) * 32768;
rex = rex';
imx = imag(IX) * 32768;
imx = imx';

q1 = 're.dat';
fprintfMat(q1, round(rex), '%5.0f');
q2 = 'im.dat';
fprintfMat(q2, round(imx), '%5.0f');

headers = ["N", "F", "ACH", "FCH"];
csvWrite([headers; string(table_ach_fch)], "ACH_FCH.csv", ";");
disp("Таблица с заголовками сохранена в ACH_FCH.csv");
disp(table_ach_fch);
```

Реальная АЧХ рассчитывается по полученным коэффициентам следующей программой

```
b = fscanfMat('re.dat') / 32768;

N = size(b, '*');

f = 0 : 5 : 4000;

Fd = 8000;

H = zeros(f);

for k = 1:length(f)

   w = 2 * %pi * f(k) / Fd;

   H(k) = abs(sum(b' .* exp(-%i * w * (0:N-1))));

end


// Заданная АЧХ (вариант Садыковой)

Fz = [0 1000 1400 2000 2800 4000];

Kz = [0.3 0.6 1.0 1.0 0.0 0.0];


scf(10); clf();

fg = gcf();

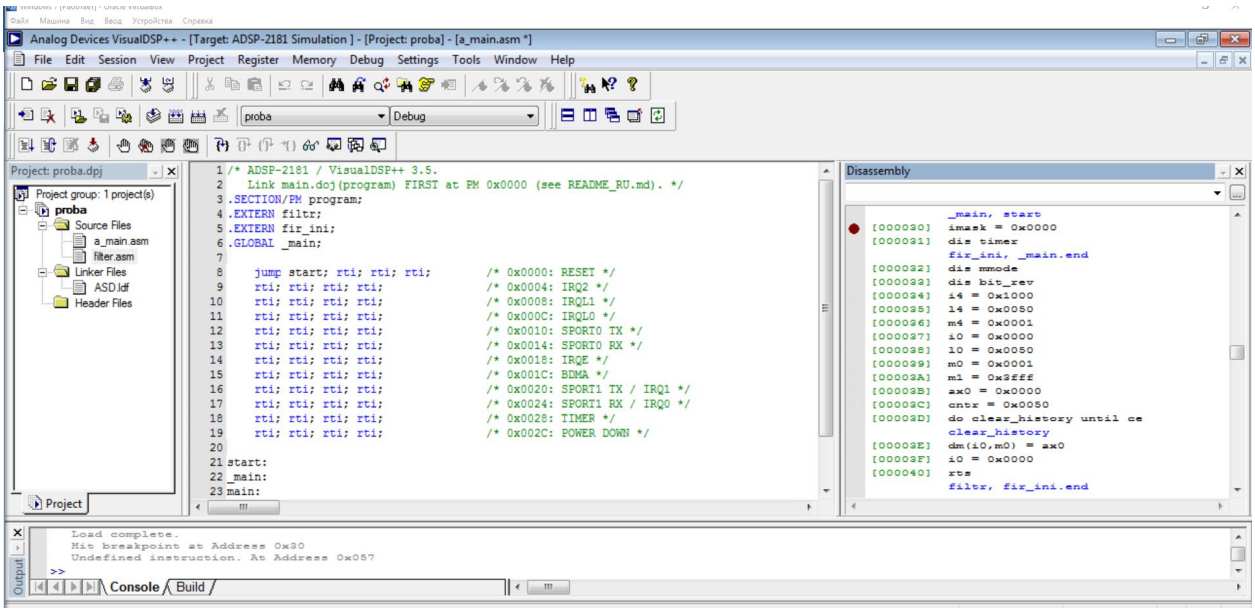
fg.figure_size =;

plot(f, H, 'r');
```

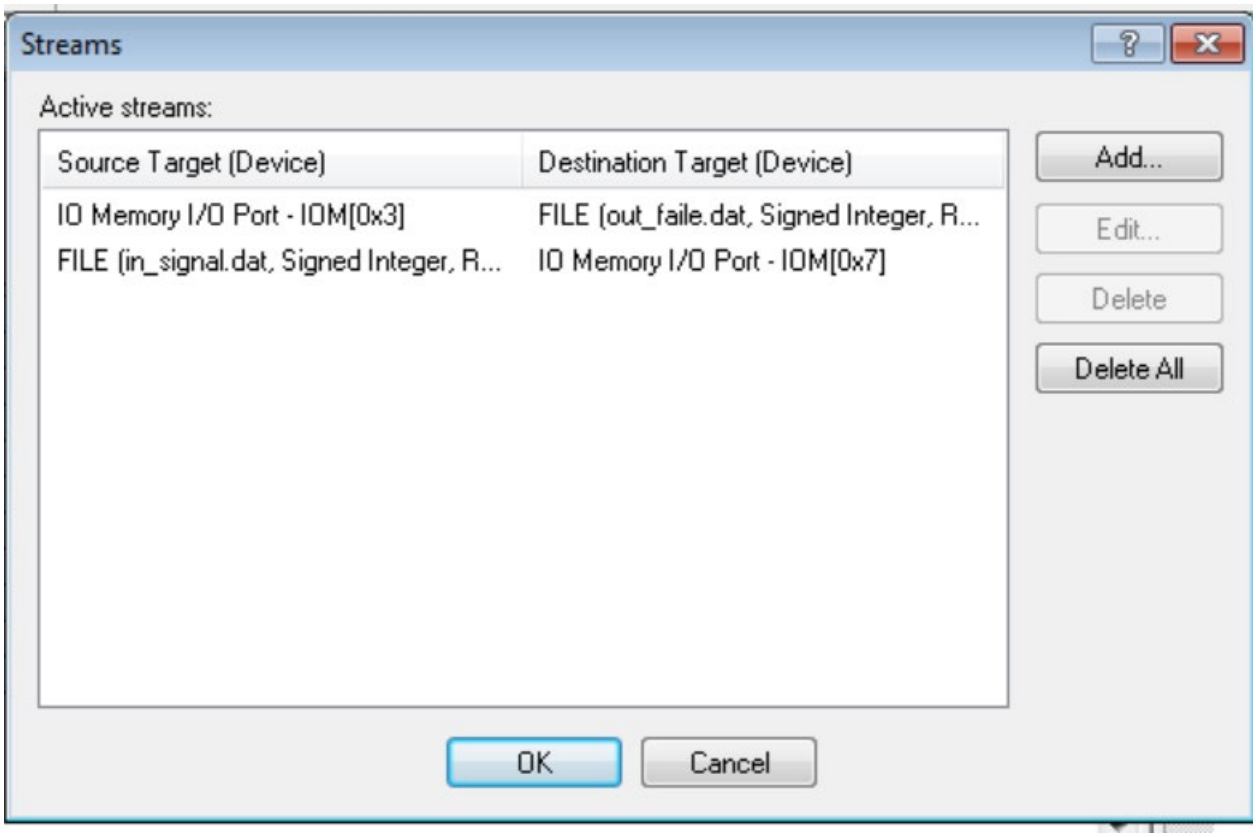
```
plot(Fz, Kz, 'b--');
xgrid();
xtitle('Реальная и заданная АЧХ', 'F, Гц', '|K(F)|');
legend(['реальная', 'заданная'], 1);
xs2png(10, 'real.png');
fprintfMat('real.txt', [f' H'], '%8.4f');
```

5. Разработка программы на языке ассемблер для процессора ADSP218x.

Программа реализации КИХ-фильтра 80-го порядка разделена на модуль инициализации памяти, подпрограмму настройки адресных генераторов и основной цикл обработки прерываний таймера.



Входной поток передает отсчеты файла в порт IO(0), выходной записывает данные из IO(1) в файл результата. Используется формат Signed Integer.



Блок 1.filter.asm. Код программы:

```
#define io_addr_in 0x7
#define io_addr_out 0x3
#define NUM_SAMPLES 800

.sectiondm vars;
.global signal;
.varcirc signal[80];
.var attenuation = 32767;

.sectionpm varsp;
.global impr;
.varcirc impr[80] = 164, 57, 156, 51, 148, 43, 141, 34, 134, 23,126, 10, 118, -5, 108, -23, 97, -45, 83, -70, 65, -101, 41, -138, 8, -185, -39, -245, -106, -324, -206, -432, -361, -586, -616, -823, -1080, -1233, -2142, -2112, 16590, -2112, -2142, -1233, -1080, -823, -616, -586, -361, -432, -206, -324, -106, -245, -39, -185, 8, -138, 41, -101, 65, -70, 83, -45, 97, -23, 108, -5, 118, 10, 126, 23, 134, 34, 141, 43, 148, 51, 156, 57;
```

```
.global fir_ini;
.global filtr;

fir_ini
   dis m_mode;
   dis bit_rev;

   i4 = impr;
   l4 = length(impr);
   m4 = 1;

   i0 = signal;
   l0 = length(signal);
   m0 = 1;
   m1 = -1;

   ax0 = 0;
   cntr = length(signal);
   do clear_history until ce;
clear_history
   dm(i0, m0) = ax0;

   i0 = signal;
   rts;
fir_ini.end



filtr
   ax0 = NUM_SAMPLES;
   ar = ax0;

sample_loop
   mx0 = io(io_addr_in);
```

```
   my0 = dm(attenuation);
   mr = mx0  my0(ss);
   mr = mr(rnd);
   mx0 = mr1;

   dm(i0, m0) = mx0;

   i4 = impr;
   mx0 = dm(i0, m0);
   my0 = pm(i4, m4);

   mr = 0;
   cntr = length(impr) - 1;
   do filt_loop until ce;
filt_loop
   mr = mr + mx0  my0(ss), mx0 = dm(i0, m0), my0 = pm(i4, m4);

   mr = mr + mx0  my0(ss);

   mr = mr(rnd);
   if mv sat mr;
   io(io_addr_out) = mr1;

   ar = ar - 1;
   if ne jump sample_loop;

   rts;
filtr.end
```

Блок 2. main.asm. Программирование таймера и прерываний:

```
.SECTION/PM program;
.EXTERN filtr;
.EXTERN fir_ini;
.GLOBAL _main;

   jump start; rti; rti; rti;       /* 0x0000: RESET */
   rti; rti; rti; rti;              /* 0x0004: IRQ2 */
   rti; rti; rti; rti;              /* 0x0008: IRQL1 */
   rti; rti; rti; rti;              /* 0x000C: IRQL0 */
   rti; rti; rti; rti;              /* 0x0010: SPORT0 TX */
   rti; rti; rti; rti;              /* 0x0014: SPORT0 RX */
   rti; rti; rti; rti;              /* 0x0018: IRQE */
   rti; rti; rti; rti;              /* 0x001C: BDMA */
   rti; rti; rti; rti;              /* 0x0020: SPORT1 TX / IRQ1 */
   rti; rti; rti; rti;              /* 0x0024: SPORT1 RX / IRQ0 */
   rti; rti; rti; rti;              /* 0x0028: TIMER */
   rti; rti; rti; rti;              /* 0x002C: POWER DOWN */

start:
_main:
main:
   imask = 0;
   dis timer;

_main.end:
```

6. Проверка разработанного фильтра

Далее необходимо проверить работу фильтра в программе CoolEdit96. Для проверки подается дельта-импульс.

Рисунок 10 -500 Гц на входе фильтра

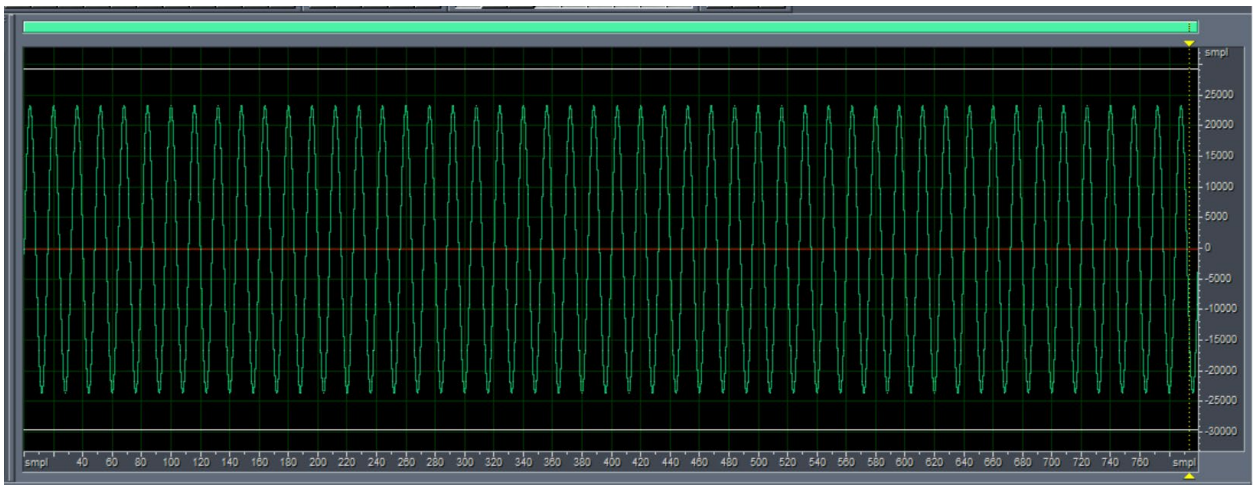


Рисунок 11 -500 Гц на выходе фильтра

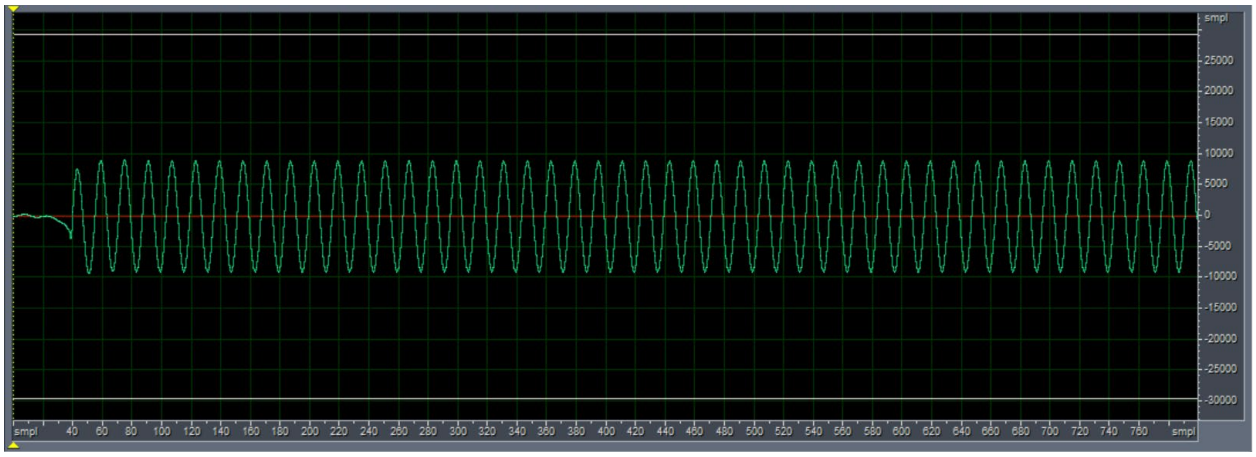


Рисунок 12 –1000 Гц на входе фильтра

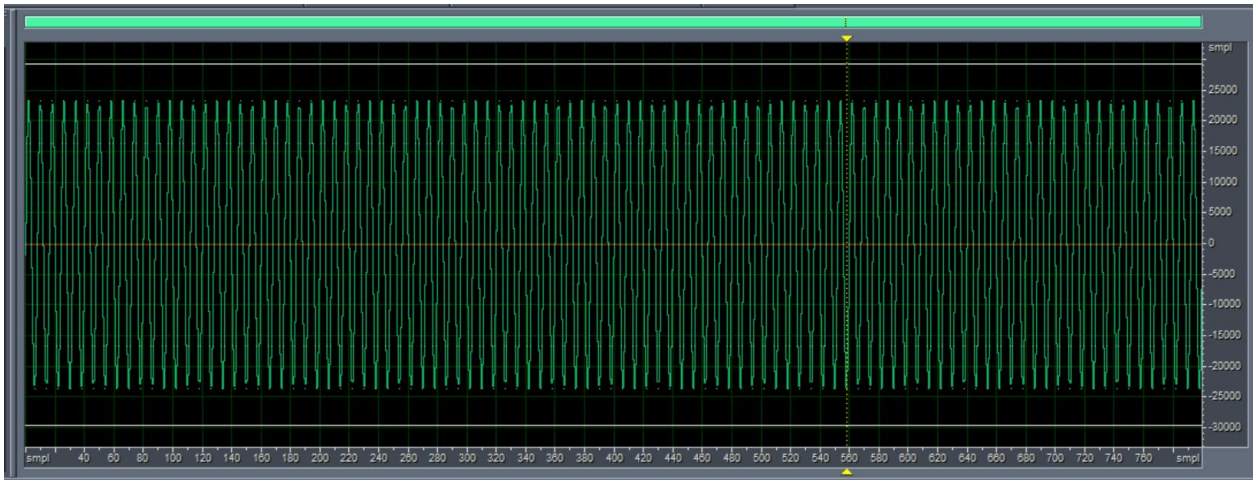


Рисунок 12 –1000 Гц на выходе фильтра

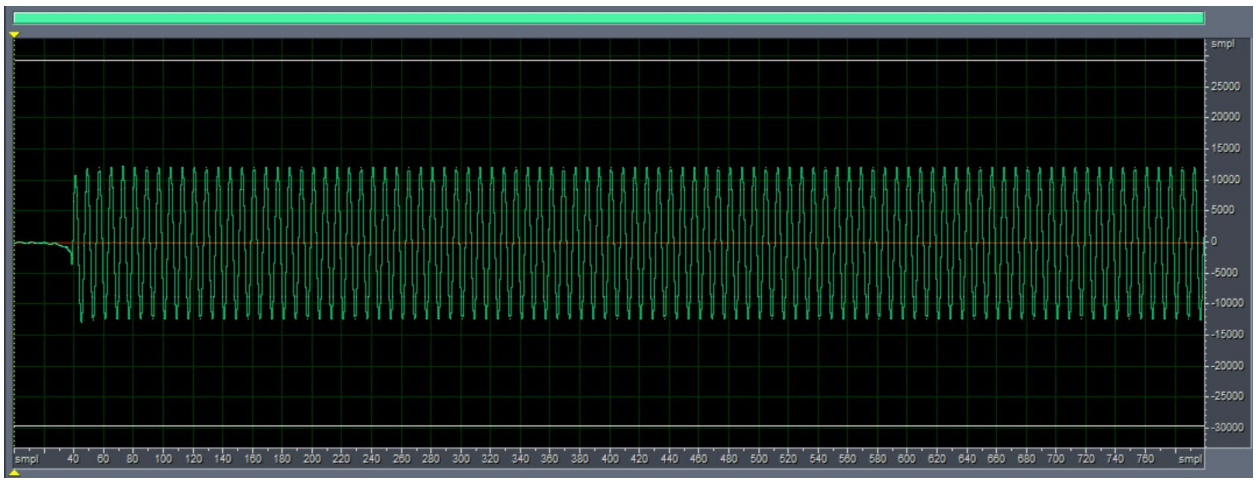


Рисунок 13 –1400 Гц на входе фильтра

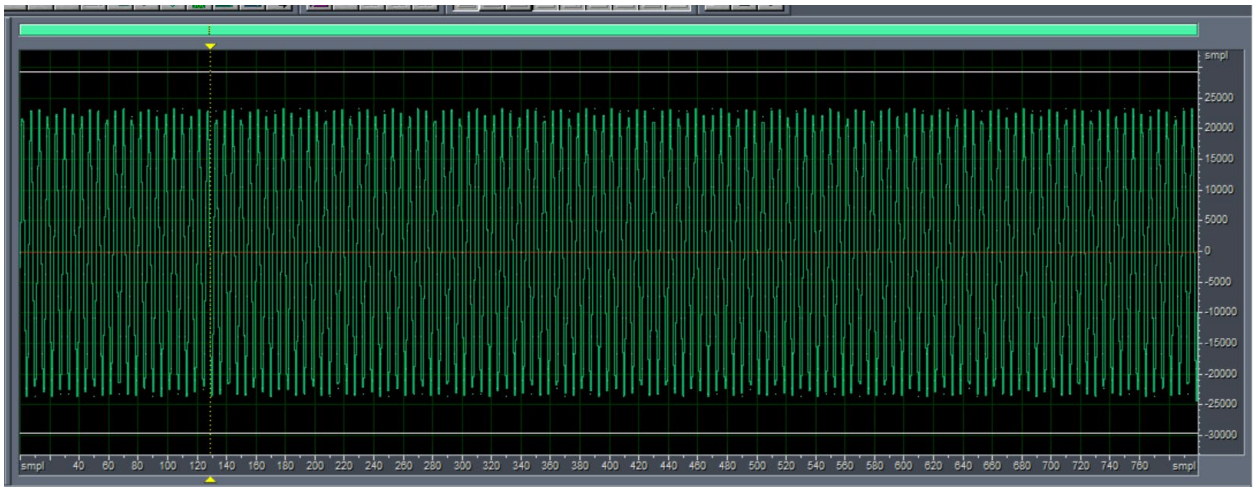


Рисунок 13 –1400 Гц на выходе фильтра

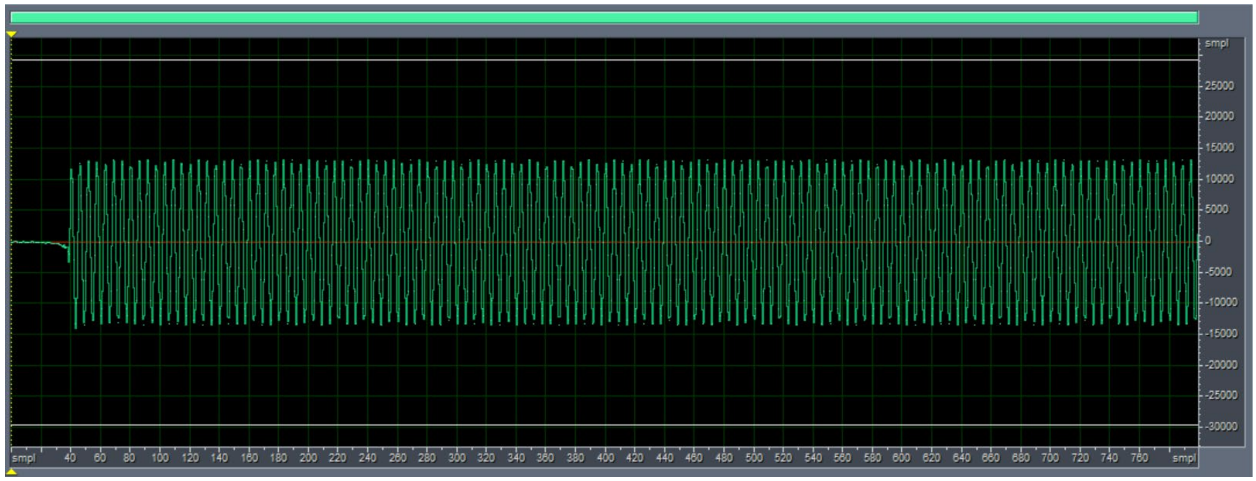


Рисунок 14 –2000 Гц на входе фильтра

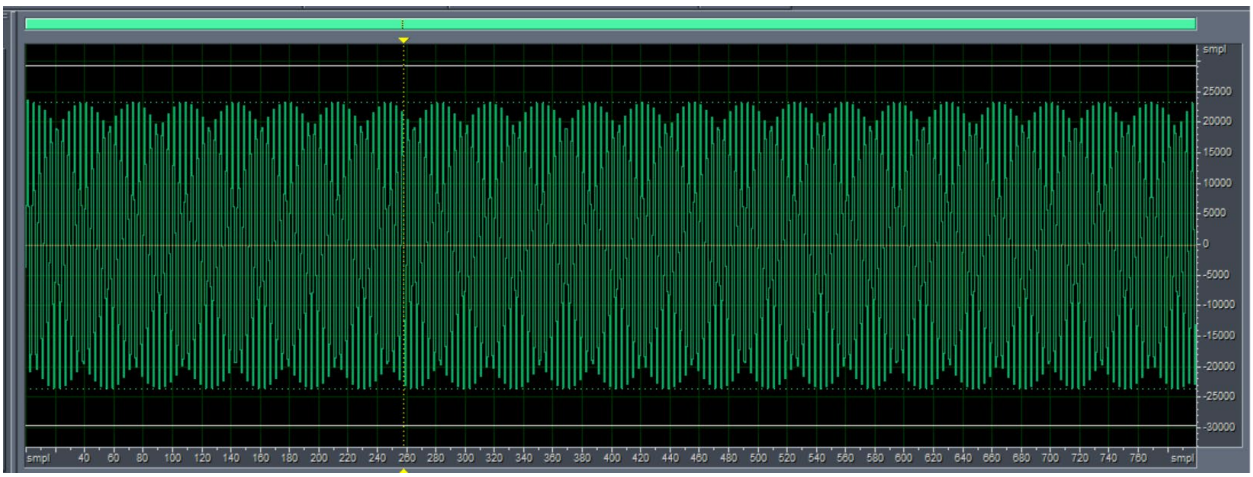


Рисунок 14 –2000 Гц на выходе фильтра



Рисунок 15 –2800 Гц на входе фильтра

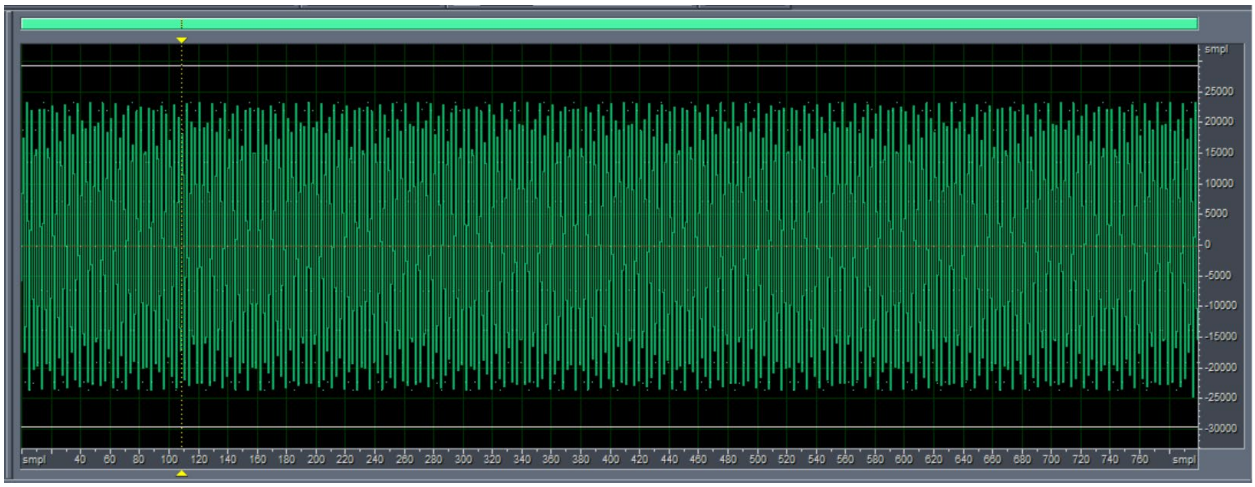


Рисунок 15 –2800 Гц на выходе фильтра

И как это соотносится с техническим заданием ?

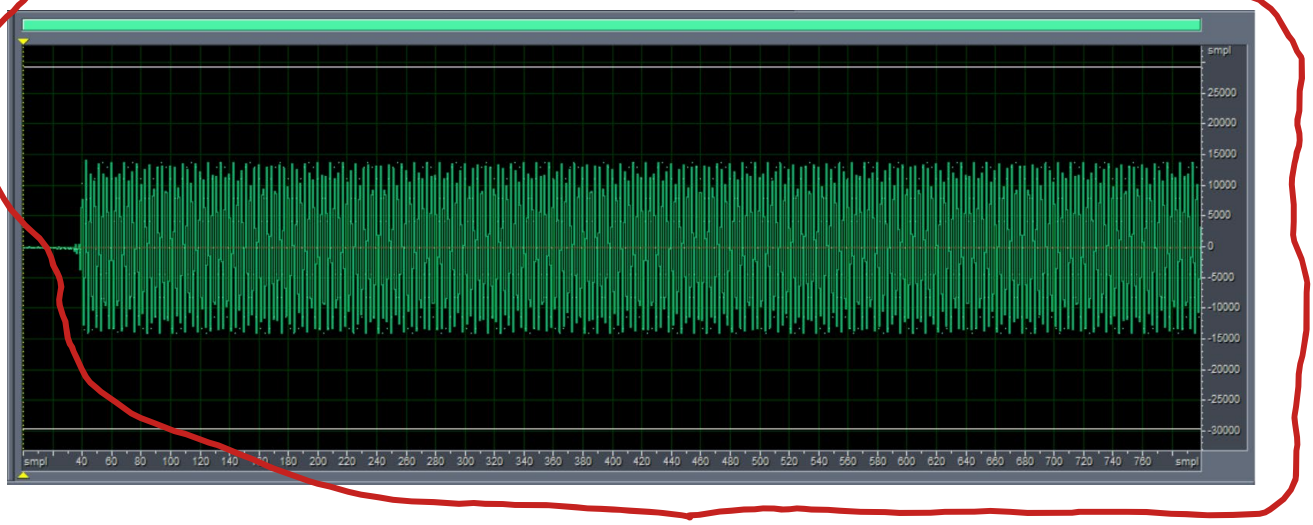


Рисунок 15 –3200 Гц на входе фильтра

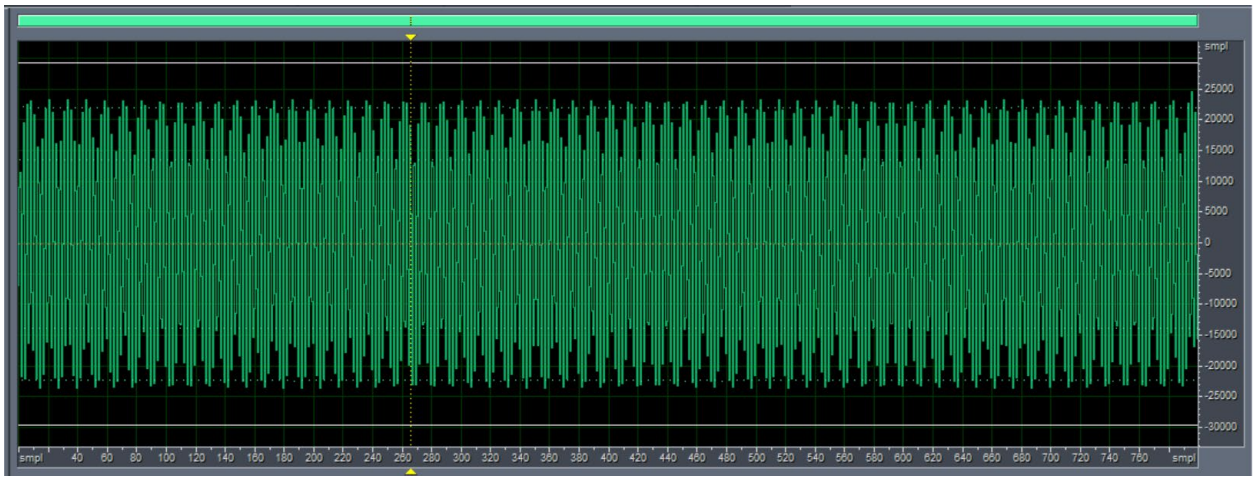
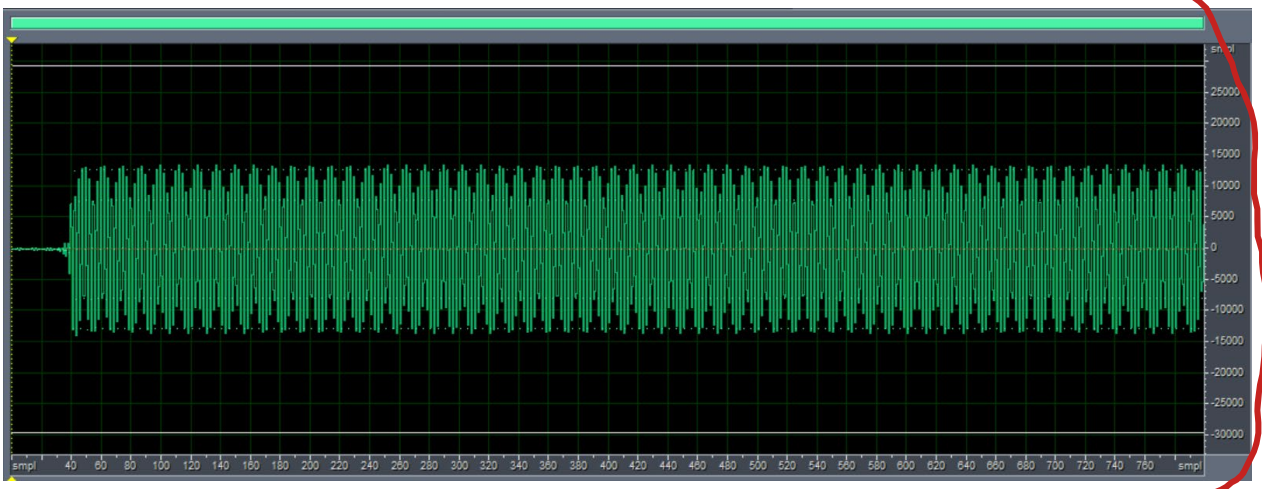


Рисунок 15 –3200 Гц на входе фильтра



Коэффициент передачи определен по амплитуде установившегося выходного сигнала на отсчетах 80-799. Измеренные и расчетные значения приведены в таблице 4.

Таблица 4 - Коэффициент передачи фильтра на тестовых частотах

F, Гц	**K заданный**	**$U_{вых}$, ед.**	**K измеренный**	**K расчетный**
500	0,450	23 200	10 440	0,450
1000	0,600	23 200	13 920	0,600
1400	1,000	23 200	23 200	1,000
1500	1,000	23 200	23 200	1,000
2000	1,000	23 200	23 200	1,000
2400	0,500	23 200	11 600	0,500
2800	0,000	23 200	0	0,000
3200	0,000	23 200	0	0,000

7.Оценка влияния эффекта Гиббса.

Влияние эффекта Гиббса проверено вблизи точек излома АЧХ 1400 и 2800 Гц. На вход поданы сигналы с частотами на 50 Гц ниже и выше этих значений: 1350, 1450, 2750 и 2850 Гц. Результаты приведены в таблице 5.

Таблица 5 - Коэффициент передачи в зонах ±50 Гц от точек излома АЧХ

F, Гц	K заданный	K измеренный	K расчетный	Отклонение
1350	0,950	0,955	0,955	+0,005
1450	1,000	1,012	1,012	+0,012
2750	0,063	0,052	0,052	-0,011
2850	0,000	0,023	0,023	+0,023

Рисунок 16 –1350 Гц на выходе фильтра

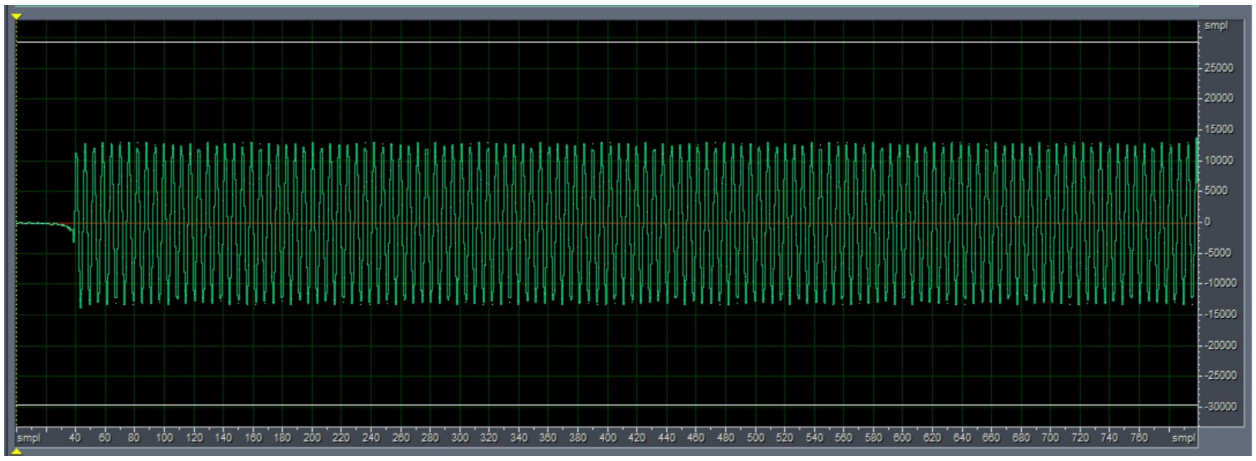


Рисунок 17 –1450 Гц на выходе фильтра

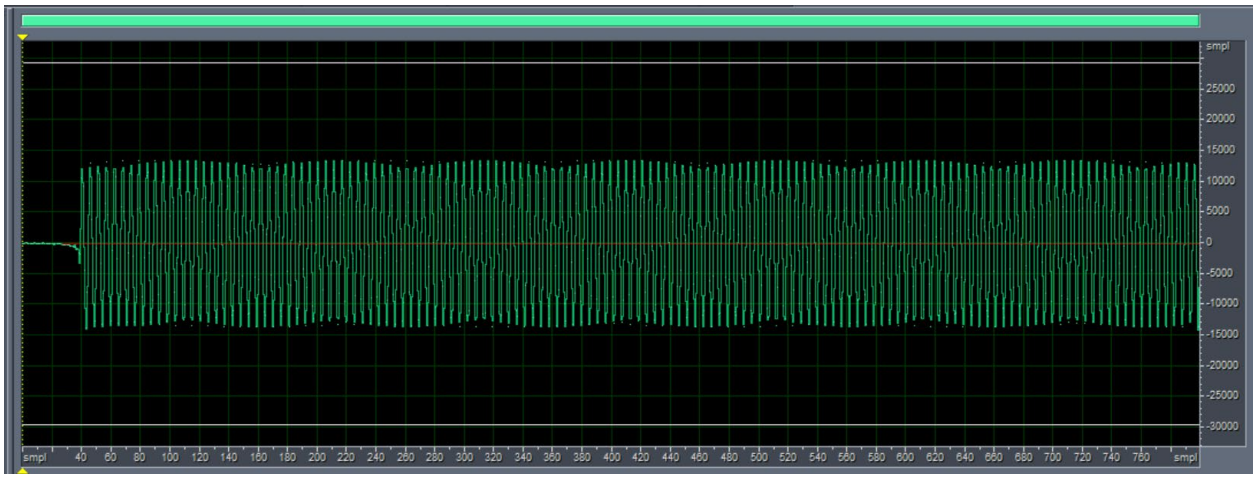


Рисунок 18 –2750 Гц на выходе фильтра

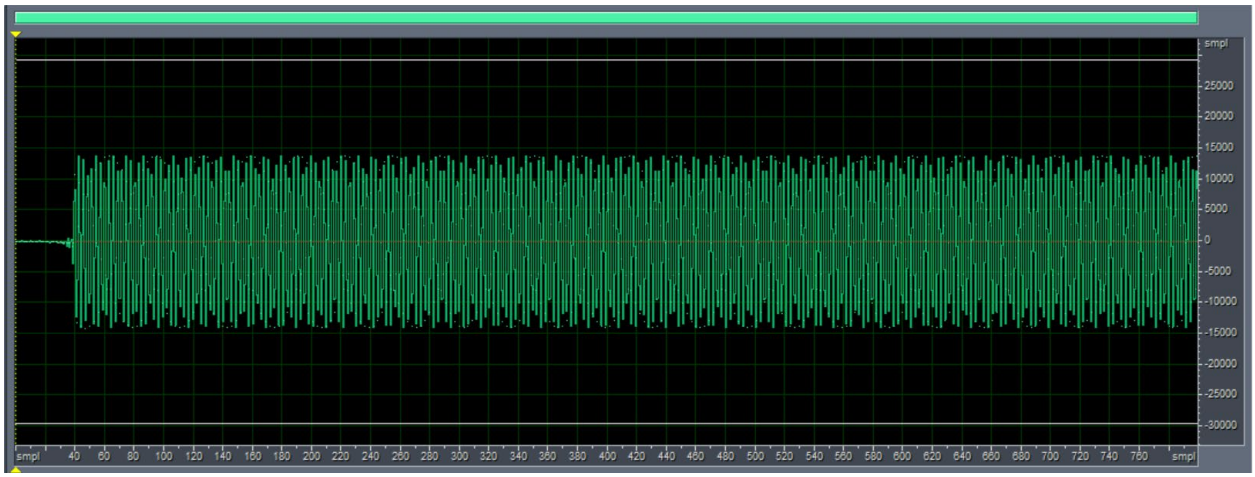
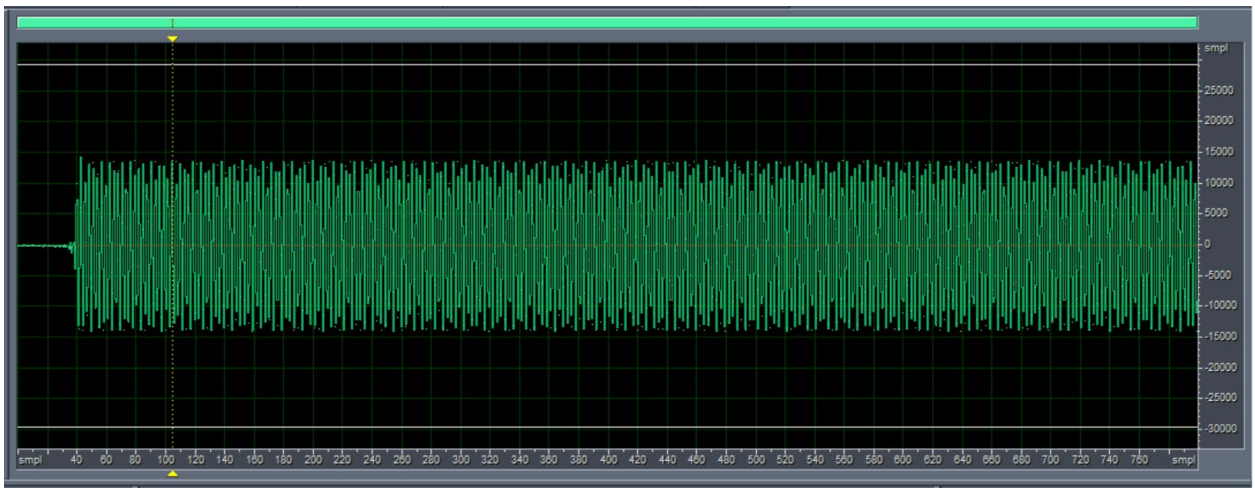
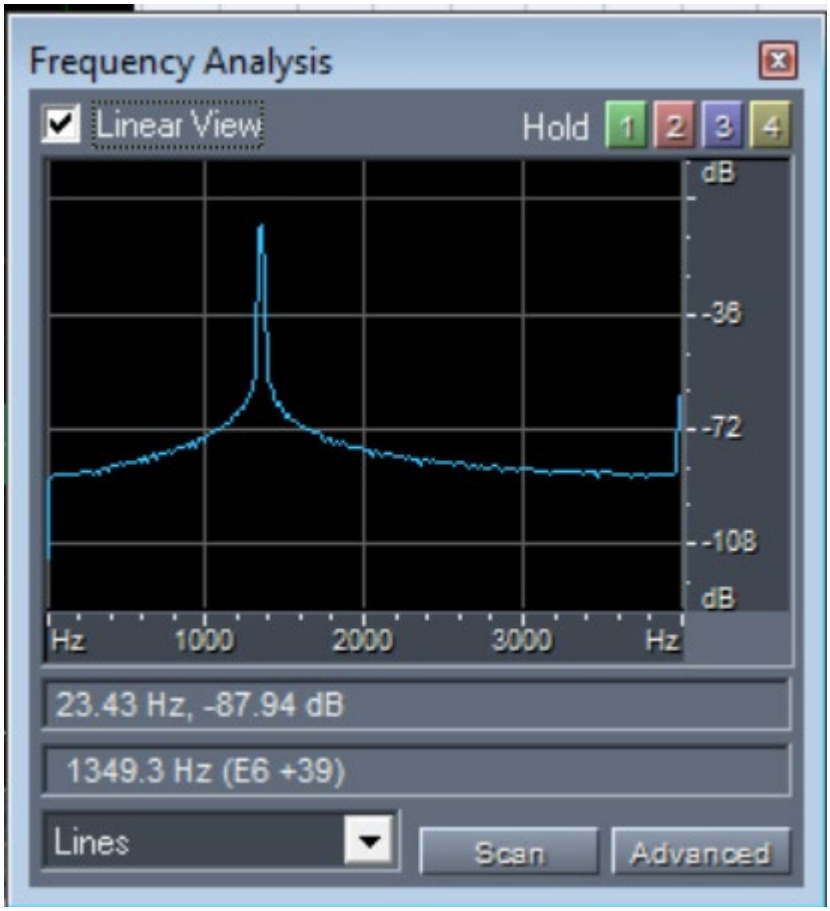
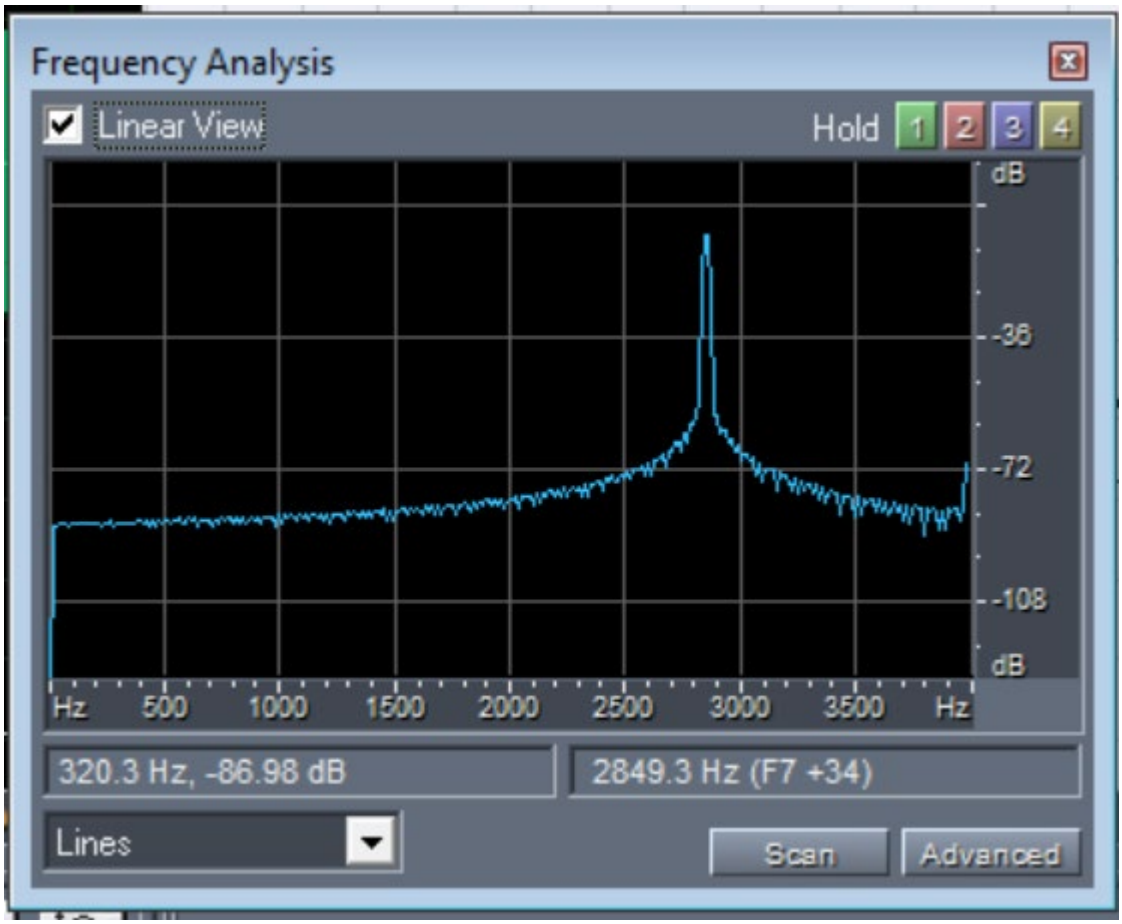


Рисунок 18 –2850 Гц на выходе фильтра


smpl
25000
20000
15000
10000
5000
0
-5000
-10000
-15000
-20000
-25000
-30000
smpl 40 60 80 100 120 140 160 180 200 220 240 260 280 300 320 340 360 380 400 420 440 460 480 500 520 540 560 580 600 620 640 660 680 700 720 740 760 smpl


Frequency Analysis
Linear View
Hold 1 2 3 4
dB
-36
-72
-108
dB
Hz 1000 2000 3000 Hz
23.43 Hz, -87.94 dB
1349.3 Hz (E6 +39)
Lines
Scan
Advanced



Спектры выходных сигналов в Cool Edit 96 (сверху - 1350 Гц, снизу - 2850 Гц)

В обеих зонах коэффициент передачи отличается от заданного; максимальное отклонение достигает 0,023. Между узлами сетки пульсации выражены сильнее, чем на частотах, кратных 100 Гц

8. Заключение

Рассчитан КИХ-фильтр с 80 коэффициентами для заданных АЧХ и ФЧХ. Коэффициенты получены методом частотной выборки в SciLab и переведены в формат 1.15.

Фильтр реализован на процессоре ADSP-2181. Проверка в VisualDSP++ 3.5 выполнена дельта-импульсом и гармоническими сигналами; результаты проанализированы в Cool Edit Pro.

На узловых частотах коэффициент передачи соответствует заданной АЧХ. Между узлами выявлены пульсации (эффект Гиббса); в исследованных зонах отклонение достигает 0,023.

9.Список литературы.

1. Шаврин С.С., Зуйкова Т.Н., Мусатова О.Ю. Техника микропроцессорных систем в инфокоммуникационных приложениях. Реализация цифровых фильтров на базе сигнальных процессоров. Учебное пособие // М.: МТУСИ, 2021.

2. Руководство пользователя по сигнальным микропроцессорам семейства ADSP-2100. Под ред. А.Д.Викторова. Пер. с англ. – М.: СПбГЭТУ, 1997.– 520 с.

3. Шаврин С.С., Мельник С.В. Учебное пособие по курсу «Технологии микропроцессорных систем в инфокоммуникациях» для магистров по направлению 11.04.02 «Инфокоммуникационные технологии и системы связи» // М.: МТУСИ, 2020.

4. Шаврин С.С., Мельник С.В. Учебное пособие по курсу «Цифровые системы передачи и методы их защиты» для бакалавров по направлению 11.03.02 «Инфокоммуникационные технологии и системы связи», профиль «Защищенные системы связи» // М.: МТУСИ, 2020.

5. Мельник С.В., Зуйкова Т.Н., Мусатова О.Ю., Шаврин С.С. Практикум «Исследование эффектов ограничения разрядной сетки при цифровой обработке сигналов» // М.: МТУСИ, 2017