МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ государственное БЮДЖЕТНОЕ

образовательное учреждение

высшего образования

«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кафедра систем сбора и обработки данных



**КУРСОВОЙ ПРОЕКТ**

**по дисциплине: «***Сигналы и системы***»**

**на тему:** **«***Цифровой режекторный эллиптический БИХ - фильтр***»**

|  |  |  |  |
| --- | --- | --- | --- |
| Выполнил:  Студент гр. «АТ-14»  «Черкасов А.Ю.»  «\_\_\_\_» 2024 г. | |  | Проверил:  Доцент  «Щетинин Ю.И.»  «\_\_\_\_» 2024 г. |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись) |  | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись) |

Новосибирск

2024

Содержание

**1.Введение3**

**2.Задание4**

**3.Краткий обзор методов проектирования цифрового фильтра5**

**4.Обоснование выбора и сущность выбранного метода6**

**5.Проектирование фильтра7**

5.1.Вычисляем граничные частоты8

5.2.Проверка условия геометрической симметрии8

5.3.Определение ширины полосы задерживания и центральной частоты полосы пропускания8

5.4.Расчет нормированного фильтра нижних частот8

5.5.Трансформирование НФНЧ в АФ 9

5.6.Трансформирование аналогового РФ в цифровой с помощью билинейного преобразования 10

**6.Реализация(структурная схема) фильтра13**

**7.Тестирование фильтра14**

7.1. Фильтрация суммы сигналов из полосы пропускания и полосы задерживания 14

7.2. Фильтрация сигнала частотой из полосы задерживания 15

7.3. Фильтрация сигнала частотой из полосы пропускания 16

**8.Программирование фильтра и оценка его быстродействия17**

**9.Заключение**1**8**

**10.Источники19**

**11.Приложения19**

# 1. Введение

Цифровая обработка сигналов как направление развития науки и техники зародилась в 1950-х годах и поначалу представляла собой довольно экзотическую отрасль радиоэлектроники, практическая ценность которой была далеко не очевидной. Однако за прошедшие пятьдесят лет благодаря успехам микроэлектроники системы цифровой обработки сигналов не только воплотились в реальность, но и вошли в нашу повседневную жизнь в виде CD- и DVD-проигрывателей, модемов, сотовых телефонов и многого другого.

Из всех задач, решаемых при цифровой обработке сигналов, наиболее важной является задача фильтрации. Фильтром называется цепь (система, устройство), обеспечивающая необходимую реакцию на заданный входной сигнал. Основное применение фильтров заключается в выделении полезного сигнала из аддитивной смеси его с шумом. То есть фильтр преобразует входной сигнал таким образом, что определенные полезные гармоники входного сигнала сохраняются в выходном сигнале, а нежелательные подавляются.

Фильтры сигналов разделяются на аналоговые и цифровые. В аналоговых фильтрах производится преобразование аналоговых (непрерывных) сигналов. Цифровой фильтр, работающий в реальном масштабе времени, оперирует с дискретными по времени данными. При этом очередной отсчет, соответствующий отклику фильтра, формируется по окончании каждого периода дискретизации. Среди цифровых фильтров, в свою очередь, выделяют два фундаментальных класса: фильтры с конечной импульсной характеристикой (КИХ-фильтры или нерекурсивные) и фильтры с бесконечной импульсной характеристикой (БИХ-фильтры или рекурсивные).

В зависимости от полосы частот прохождения сигнала фильтры подразделяются фильтры нижних частот (ФНЧ), верхних частот (ФВЧ), полосовые (ПФ) и режекторные (РФ, заграждающие).

Для аппроксимации характеристик фильтров используются специальные типы функций, которые могут быть реализованы в практических схемах. По названию аппроксимирующих функций соответствующие фильтры называю фильтрами Баттерворта, Чебышева, Бесселя, Кауэра и др.

Таким образом, в зависимости от требований, предъявляемых к выходному сигналу, используются те или иные фильтры. И в каждом конкретном случае требуется спроектировать цифровой фильтр с теми или иными характеристиками, удовлетворяющими этим требованиям.

В данной курсовой работе, проектируется цифровой эллиптический режекторный БИХ – фильтр с помощью системы MatLab R2023b и системы Visual Studio C++, используемой для оценки быстродействия фильтра.

# 2. Задание

Разработайте цифровой режекторный эллиптический БИХ - фильтр, удовлетворяющий следующей спецификации:

* Нижняя граничная частота полосы пропускания – 46 Гц,
* Нижняя граничная частота полосы задерживания – 48,5 Гц,
* Верхняя граничная частота полосы задерживания – 52 Гц,
* Верхняя граничная частота полосы пропускания – 54 Гц,
* Неравномерность передачи в полосе пропускания – 0,5 дБ,
* Минимальное ослабление в полосе задерживания – 50 дБ,

Частота дискретизации – 300 Гц.

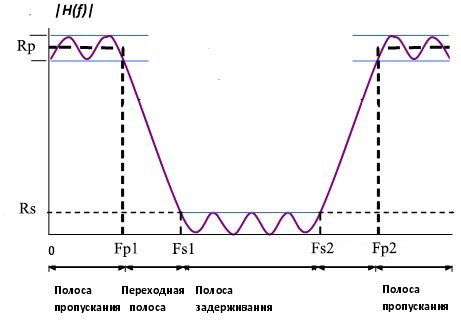


Рис.1. Вид спецификации режекторного фильтра.

# 3. Краткий обзор методов проектирования цифровых БИХ – фильтров

Цифровые БИХ фильтры описываются с помощью конечно-разностного уравнения вида:

 [1].

Расчёт фильтра заключается в определении значений коэффициентов аk и bk, обеспечивающих необходимый вид амплитудной, фазовой или импульсной характеристик.

Существует 3 группы методов расчёта БИХ-фильтров:

1. Методы, основанные на использовании аналоговых прототипов фильтров.

2. Методы оптимизации.

3. Прямые методы расчёта в z-плоскости. [1]

Для первой группы проектируется соответствующий аналоговый фильтр-прототип и дискретизируется его передаточная функция. Два основных метода: метод билинейного z-преобразования и метод инвариантной импульсной характеристики. Билинейный метод позволяет синтезировать рекурсивный дискретный фильтр по частотной характеристике аналогового прототипа. Функция передачи аналоговой цепи с сосредоточенными параметрами представляет собой дробно-рациональную функцию переменной s. Чтобы получить функцию передачи дискретного фильтра, необходимо перейти из s-области в z-область, причем дробно-рациональный характер функции должен сохраниться. Инвариантный метод позволяет синтезировать рекурсивный дискретный фильтр путем дискретизации импульсной характеристики аналогового прототипа. [3]

Для второй группы применяются сложные методы оптимизации, минимизирующие отклонение аппроксимирующей характеристики от заданной характеристики фильтра. Отличительной чертой этих методов является то, что система уравнений, составленная относительно коэффициентов фильтра, не может быть решена в явной форме. Поэтому для нахождения коэффициентов приходится использовать численные методы оптимизации, минимизирующие, согласно выбранному критерию, некоторую ошибку. [4]

Для третей группы амплитудная или импульсная характеристика цифрового фильтра аппроксимируется в z-плоскости. Это позволяет получить заданный квадрат амплитудной характеристики фильтра или заданную импульсную характеристику в z-плоскости. Эти методы не требуют использования аналогового прототипа фильтра, т.е. относятся к прямым методам проектирования ЦФ. Они позволяют определить передаточную функцию ЦФ непосредственно в z – плоскости и минимизируют разность между желаемой и действительной частотными характеристиками фильтра. [2]

# 4. Обоснование выбора и сущность выбранного метода проектирования

Наиболее распространены методы расчета БИХ-фильтров, вклю­чающие проектирование соответствующего аналогового прототипа, т.к. разработка аналоговых прототипов раньше предшествовала цифровым фильтрам и их методы расчета хо­рошо известны. Этот метод проектирования можно разделить на этапы:

- используя спецификации цифрового фильтра, необходимо получить спецификации аналогового фильтра;

- проектирование АФ, а именно нормированного фильтра нижних частот;

- переход от НФНЧ к цифровому фильтру.

Такой переход от НФНЧ к режекторному ЦФ осуществляется путем введения замены типа

, где - центральная полоса и

– ширина полосы задерживания, где и – нижняя и верхняя граничные угловые частоты полосы пропускания, и – граничные угловые частоты нижней и верхней полос задерживания. [1]

Синтез аналогового фильтра включает следующие этапы:

1. На основе спецификаций исходного проектируемого ЦФ выбирается подходящая аппроксимация для АЧХ: аппроксимация Баттерворта или Чебышева или эллиптическими функциями.
2. С помощью преобразования где - частота дискретного фильтра, а - интервал дискретизации, определяются граничные угловые частоты полос задерживания и пропускания аналогового фильтра. Необходимость пересчитать частоты возникает из-за нелинейности преобразования.

3. Проектируется аналоговый нормированный ФНЧ. Определяется порядок и передаточная функция НФНЧ. [1].

Переход от полученного аналогового прототипа фильтра к цифровому может осуществляться различными методами:

1. Билинейным преобразованием.

2. Методом инвариантного преобразования импульсной характеристики аналогового фильтра в цифровой.

3. Методом дискретизации на основе дискретизации дифференциального уравнения аналогового фильтра соответствующим линейным разностным уравнением.

В данной работе применяется метод, основанный на использовании аналогового прототипа, а переход к цифровому фильтру осуществляется на основе метода билинейного преобразования. Этот метод преобразования аналогового фильтра в цифровой применяется наиболее часто ввиду его простоты и качеств результирующих характеристик фильтров.

Метод билинейного преобразования:

Передаточная функция H(s) «s-плоскости» преобразуется в передаточную функцию H(z) «z-плоскости» цифрового БИХ-фильтра с помощью функции:

[1]

Устойчивость полученного таким образом цифрового фильтра гарантируется, если исходный аналоговый фильтр был устойчив и область устойчивости в одной комплексной плоскости путем преобразования отображается в область устойчивости в другой комплексной плоскости. Достоинством метода билинейного преобразования является то, что передаточная функция цифрового фильтра определяется с помощью простых формул из передаточной функции аналогового фильтра, для которых существуют подробные таблицы и справочники. Также метод билинейного преобразования по сравнению с другими методами преобразования аналоговых фильтров в цифровые (инвариантности импульсной характеристики и согласованного z-преобразования) обеспечивает построение такого БИХ-фильтра, выходной, сигнал которого приближенно совпадает с выходным сигналом аналогового фильтра-прототипа при произвольных одинаковых входных сигналах. Метод билинейного преобразования имеет и свои недостатки, наиболее значимым из которых является несоответствие импульсной и фазовой характеристик аналогового и цифрового фильтра.

# 5. Проектирование фильтра

* Нижняя граничная частота полосы пропускания – 46 Гц,
* Нижняя граничная частота полосы задерживания – 48,5 Гц,
* Верхняя граничная частота полосы задерживания – 52 Гц,
* Верхняя граничная частота полосы пропускания – 54 Гц,
* Неравномерность передачи в полосе пропускания – 0,5 дБ,
* Минимальное ослабление в полосе задерживания – 50 дБ,

Fp1=46;

Fp2=54;

Fs1=48.5;

Fs2=52;

Rs=50;

Rp=0,5;

Ft=300;

5.1. Вычисляем граничные частоты по формуле (Приложение 1)

wp1 = 333.9557

wp2 = 363.3729

ws1 = 313.6724

ws2 = 380.7716

5.2. Проверка условия геометрической симметрии, которое требуется для данного метода

wp1\*wp2= 121350.451

ws1\*ws2= 119437.541

Поскольку Ws1\*Ws2 ≠ Wp1\*Wp2, то скорректируем частоту

ws1=wp1\*wp2/ws2= 318,6962

5.3. Определение ширины полосы задерживания и центральной частоты полосы пропускания (Приложение 1)

Bw=29.4172

Wo=348.3539

5.4 Расчет нормированного фильтра нижних частот (Приложение 2)

Граничной частота полосы задерживания нормированного фильтра:

Для нормированного ФНЧ частота среза = 1, а граничная частота полосы задерживания (wp2-wp1) = 2.1102 [1]

Определим минимальный порядок фильтра N, с помощью функции

ellipord(1, 2.1102,Rp,Rs,’s’) [5]

N = 4.

Выполним расчет передаточной функции с помощью функции

ellip(N, Rp,Rs,1,'low','s')[6]

Получим коэффициенты:

num1 =

0.003162277660013 0.000000000000000 0.098308715709345 0.000000000000000 0.407393003840062

den2 =

1.000000000000000 1.186532031488904 1.737465579648977 1.060836173598294 0.431532556928752

5.5. Трансформирование НФНЧ в АФ ( Приложение 3)

Для преобразования НФНЧ в режекторный аналоговый фильтр используем функцию lp2bs(num1,den1, Wo, Bw).[1]

Получим коэффициенты:

num2 =1.0e+20 \*

0.000000000000000 0.000000000000000 0.000000000000005 0.000000000000000 0.000000000834609 0.000000000000000 0.000067510464288 0.000000000000000 2.047225778437960

den2 = 1.0e+20 \*

0.000000000000000 0.000000000000000 0.000000000000005 0.000000000000264 0.000000000892030 0.000000032032685 0.000071993043115 0.001292291232362 2.168531532089396

Построим графики АЧХ и ЛАЧХ аналогового РФ:

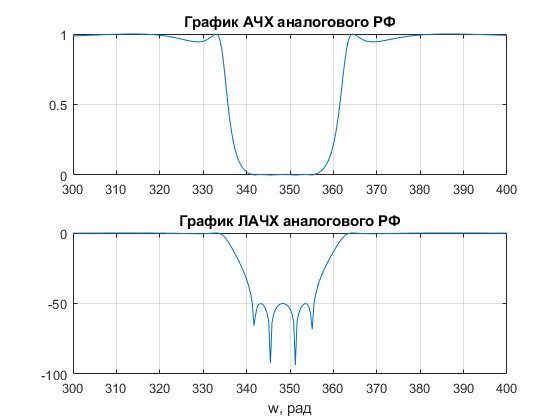


Рис.3. Графики АЧХ и ЛАЧХ аналогового режекторного фильтра.

5.6.Трансформирование аналогового РФ в цифровой с помощью билинейного преобразования(Приложение 4)

Воспользуемся функцией bilinear(num2, den2, Ft), чтобы получить коэффициенты передаточной функции исходного цифрового фильтра. [1]

Получим коэффициенты:

numFil = 0.8618 -3.4179 8.5297 -13.6118 16.1675 -13.6118 8.5297 -3.4179 0.8618

denFil = 1.0000 -3.8745 9.4466 -14.7337 17.1070 -14.0846 8.6327 -3.3848 0.8352

5.6.1 Построение графиков

Построим график АЧХ и ЛАЧХ для искомого ЦФ:

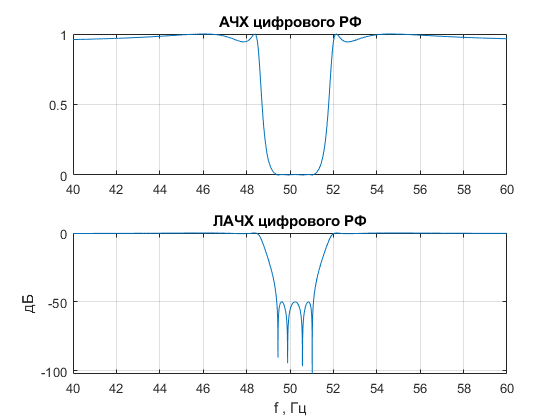


Рис.4. Графики АЧХ и ЛАЧХ цифрового режекторного фильтра.

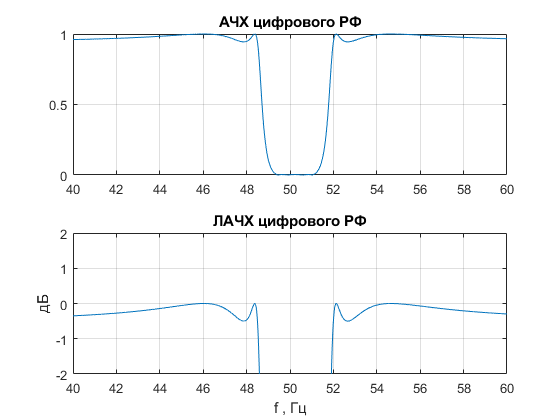
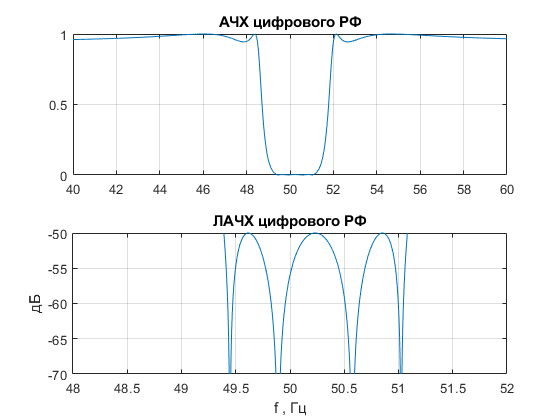


Рис.5. Приближенные графики АЧХ и ЛАЧХ цифрового режекторного фильтра.

Как видно из рис.5 фильтр удовлетворяет заявленной спецификации.

Построим диаграмму нулей и полюсов:

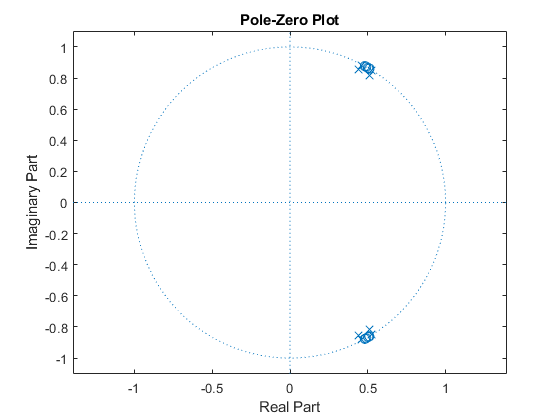
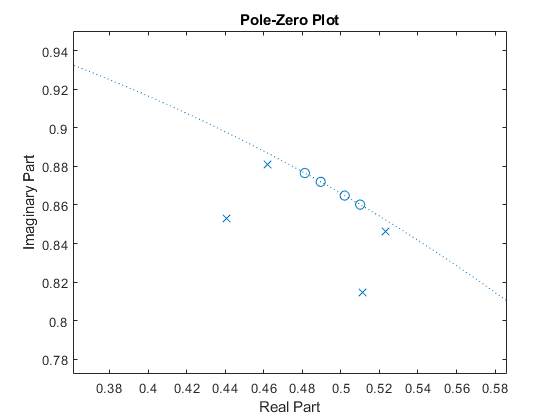
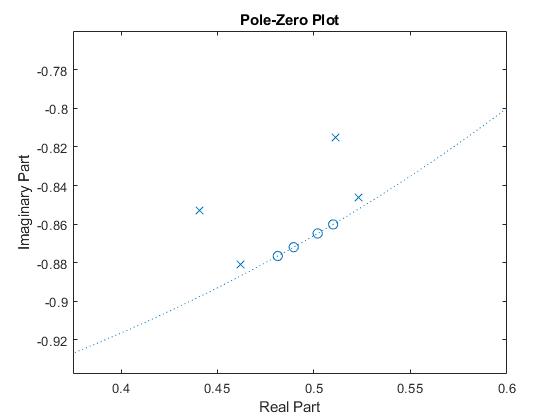


Рис.6. Диаграмма нулей и полюсов цифрового режекторного фильтра.

Из диаграммы видно, что полюса лежат внутри единичной окружности, значит система устойчива.

Построим график импульсной характеристики и ФЧХ:

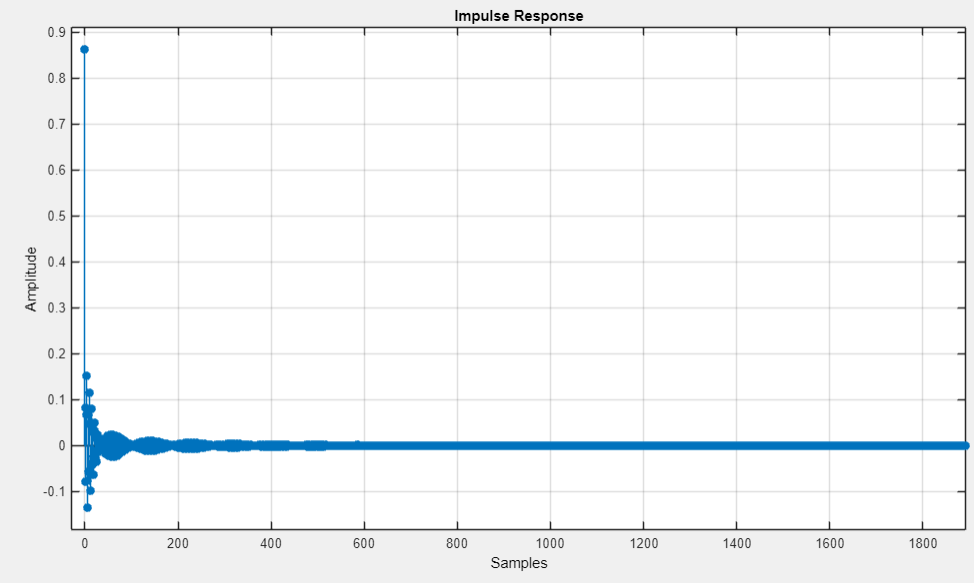


Рис.7. Импульсная характеристика цифрового режекторного фильтра.

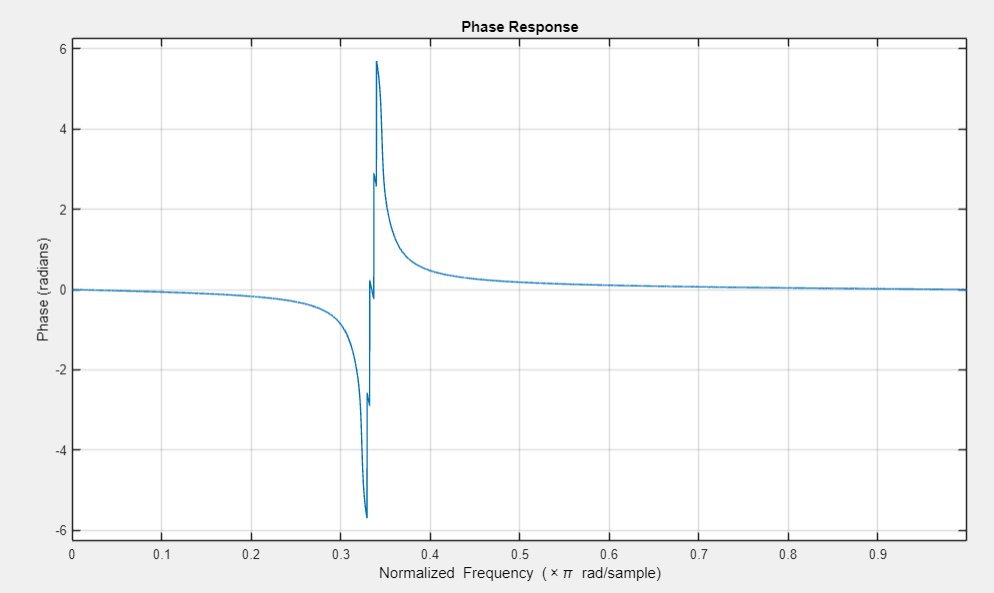


Рис.8. ФЧХ цифрового режекторного фильтра.

Исходя из рис.7 и рис.8 видно, что импульсная характеристика бесконечная, а ФЧХ нелинейна, значит это действительно БИХ-фильтр. [1]

# 6. Реализация (структурная схема) фильтра

Последовательная форма реализации цифрового фильтра менее чувствительна к погрешностям квантования, возникающим при округлении коэффициентов, поэтому выбрана именно она. Полученный фильтр можно реализовать как каскадное соединение четырех фильтров второго порядка.

Каскадная форма представления передаточной функции данного фильтра имеет вид:

H(z) = H1(z)\*H2(z)\*H3(z)\*H4(z)

Строки матрицы *sos* содержат коэффициенты фильтров второго порядка.

sos = tf2sos(numFil, denFil) (Приложение 5)

sos =

0.861888691792640 -0.843859193013008 0.861888693061288 1.000000000000000 -0.881800479426981 0.921806264502153

1.000000000000000 -1.003983264363036 0.999999998511189 1.000000000000000 -1.022392215395558 0.925293847764276

1.000000000000000 -0.962613542903116 0.999999999366148 1.000000000000000 -0.924328187096459 0.989345165082282

1.000000000000000 -1.020011642567023 1.000000000650734 1.000000000000000 -1.046054115243146 0.989762247489705

Запишем каскадную форму передаточной функции:

Построим график каскадной формы:

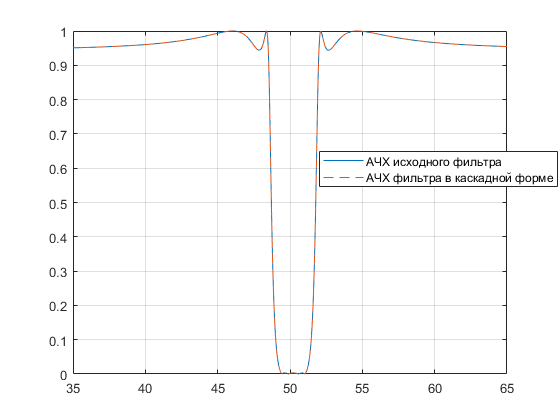


Рис. 9. АЧХ спроектированного фильтра и его каскадной формы.

По графику видно, что АЧХ спроектированного фильтра и его каскадной формы практически идентичны, следовательно, такая реализация фильтра допустима. Каскадная реализация необходима для возможности физического создания фильтра. Она, как было сказано ранее, более устойчива к ошибкам квантования и погрешностям округления по сравнению с прямой формой.[1]

# 7. Тестирование фильтра

Для проверки работоспособности фильтра подадим на его вход сигнал, состоящий из суммы двух гармоник. Частота одной из них(50Гц) находится в полосе задерживания фильтра, а частота другой(70Гц) – в полосе пропускания. (Приложение 6) [1]

Так как фильтр у нас выражен в каскадной форме, то воспользуемся функцией filter() несколько раз, то есть отфильтруем сигнал каждым из четырех фильтров 2-го порядка.

7.1. Фильтрация суммы сигналов из полосы пропускания и полосы задерживания:

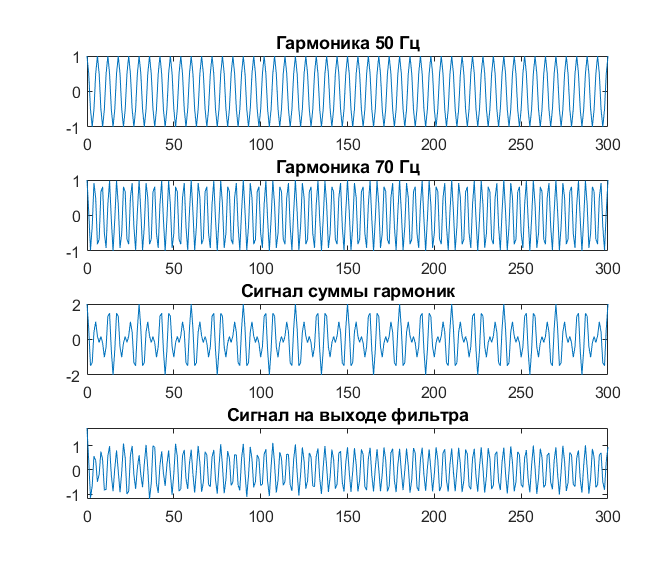


Рис. 10. Сигналы двух гармоник, их суммы и выходной сигнал.

По графику видно, что фильтр пропускает только гармонику с частотой, лежащей в полосе пропускания, и подавляет гармонику с частотой из полосы задерживания. Количественные оценки фильтрации сигналов приведены ниже.

7.2. Фильтрация сигнала частотой из полосы задерживания y2 =cos(2\*pi\*50\*(1/Ft)\*n):

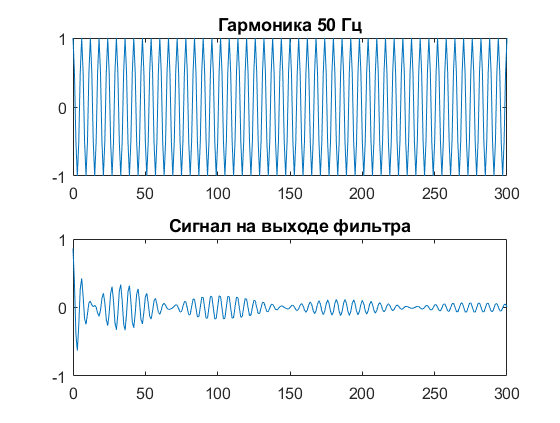


Рис. 10. Сигнал с частотой 50 Гц и сигнал на выходе фильтра.

По графику видно, что фильтр ослабляет сигнал из полосы задерживания. Ослабление сигнала удовлетворяет заданным требованиям:

20\*log10((y2)/(F4)) = -50.286876155147624 Дб

7.3. Фильтрация сигнала частотой из полосы пропускания y3 =cos(2\*pi\*70\*(1/Ft)\*n):

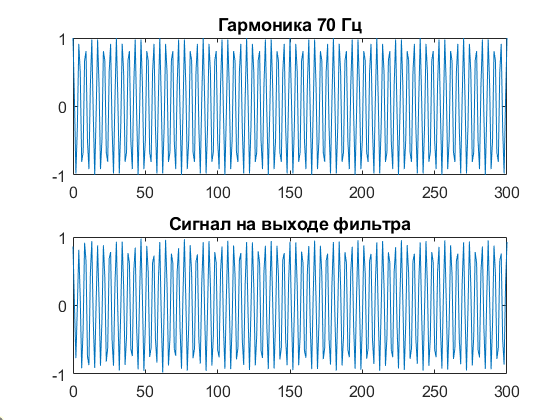


Рис. 10. Сигнал с частотой 70 Гц и сигнал на выходе фильтра.

По графику видно, что сигнал с частотой из полосы пропускания проходит полностью лишь с ослаблением в пределах заданных спецификацией:

20\*log10(max(y3)/max(F4)) = 0.208833194934224 Дб

# 8. Программирование фильтра и оценка его быстродействия

Для программирования фильтра и анализа его быстродействия полученного фильтра используется программа на языке С++, позволяющая определить время, за которое программный фильтр обрабатывает входной сигнал с заданным количеством отсчетов. (Приложение 7)

Чтобы измерить время вычисления, мы использовали функцию GetTickCount(), чтобы получить системное время в миллисекундах. Эта функция вызывается перед и после вычисления выходного сигнала, а затем мы определяем время фильтрации сигнала, вычислив разницу между двумя значениями. Метод приводит приблизительное время вычисления цикла, которое зависит от конфигурации и доступных ресурсов системы, на которой запущена программа.

|  |  |  |
| --- | --- | --- |
| Количество  отсчетов (N) | Общее время (t), мс. | Время обработки  одного отсчета (Т), мкс |
| 5 000 | <1 | 0.2 |
| 50 000 | 7.9 | 0.158 |
| 100 000 | 16 | 0.16 |

Табл.1. Оценка быстродействия для разного количества отсчетов



Рис. 11. Замер времени для 100000 отсчетов.

На компьютере с процессором AMD Ryzen 5 5600G OEM с нормальной тактовой частотой 3.9 ГГц время фильтрации входного сигнала, являющегося суммой гармоник частотой 50 и 70 Гц при количестве отсчетов N=100000 составило 16мс = 0.016 секунды, следовательно, один отсчет выполняется за 0.16 мкс. Максимальная частота, с которой способен работать цифровой фильтр, обратно пропорциональна времени обработки одного отсчета, т.е. fmax= 6.25 МГц.

# 9. Заключение

В результате курсовой работы был спроектирован режекторный эллиптический БИХ-фильтр, удовлетворяющий требованиям спецификации. При проектировании применялся метод аналогового прототипа, а переход к цифровому фильтру был осуществлен с помощью метода билинейного преобразования. Реализация полученного фильтра выполнена в форме последовательного соединения чертырех каскадов 2-го порядка. Тестирование фильтра показало его работоспособность: он отфильтровывает сигналы с частотами из полосы задерживания и пропускает сигналы с частотами из полосы пропускания. Быстродействие фильтра составляет 0,16 мкс на отсчет, что даёт предельную рабочую частоту 6.25 МГц.

# 10.Источники

1. Щетинин Ю. И. Лекция №13 «Цифровые фильтры. Введение.».: Новосибирск: 7-й учебный корпус НГТУ. Сеть/ssodserver/study/Сигналы и системы\_2024/Лекции/Лекция №14

2. Щетинин Ю. И. Лекция №13 «Цифровые фильтры. Введение.».: Новосибирск: 7-й учебный корпус НГТУ. Сеть/ssodserver/study/Сигналы и системы\_2024/Лекции/Лекция №16

3. Сергиенко А.Б. Цифровая обработка сигналов. - СПб.: БХВ-Петербург, 2011.

4. [Балашов Е.П. и др. Микро- и мини-ЭВМ / Е.П. Балашов, В.Л. Григорьев, Г.А. Петров: Учебное пособие для вузов. – Л.: Энергоатомиздат. Ленингр. отд-ние, 1984 [Электронный ресурс] // studizba.com информационный портал 2010-2024 URL:https://studizba.com/files/show/doc/129433-1-63943.html](Балашов%20Е.П.%20и%20др.%20Микро-%20и%20мини-ЭВМ%20/%20Е.П.%20Балашов,%20В.Л.%20Григорьев,%20Г.А.%20Петров:%20Учебное%20пособие%20для%20вузов.%20–%20Л.:%20Энергоатомиздат.%20Ленингр.%20отд-ние,%201984%20%5bЭлектронный%20ресурс%5d%20//%20studizba.com%20информационный%20портал%202010-2024%20%20https://studizba.com/files/show/doc/129433-1-63943.html) (дата обращения 24.05.2024)

5. Документация функции ellipord [Электронный ресурс]// mathworks.com официальный сайт MATLAB с документацией 2005-2024. URL:<https://www.mathworks.com/help/signal/ref/ellipord.html> (дата обращения 24.05.2024)

6. Документация функции ellip [Электронный ресурс]// mathworks.com официальный сайт MATLAB с документацией 2005-2024. URL: <https://www.mathworks.com/help/signal/ref/ellip.html> (дата обращения 24.05.2024)

# 11. Приложения

**Приложение 1.** Расчёт граничных угловых частот полос задерживания и пропускания аналогового фильтра.

Fpass1=46;

Fpass2=54;

Fstop1=48.5;

Fstop2=52;

Rs=50;

Rp=0.5;

Ft=300;

wp1=2\*pi\*Fpass1/Ft;

wp2=2\*pi\*Fpass2/Ft;

ws1=2\*pi\*Fstop1/Ft;

ws2=2\*pi\*Fstop2/Ft;

Ws1=2\*Ft\*tan(ws1/2);

Ws2=2\*Ft\*tan(ws2/2);

Wp1=2\*Ft\*tan(wp1/2);

Wp2=2\*Ft\*tan(wp2/2);

%Пересчет Wp1

Wp1=(Ws1 \* Ws2)/ Wp2;

Wo=sqrt(Ws1\*Ws2);

Bw=Ws2-Ws1;

**Приложение 2.** Расчёт передаточной функции НФНЧ.

Omega\_s = (ws2-ws1)/(wp2-wp1);

[N,Wn]=ellipord(1,Omega\_s, Rp, Rs,'s');

[num1,den1]=ellip(N, Rp, Rs,Wn,'low','s'); % Расчет перед. функции НФНЧ

**Приложение 3.** Расчёт передаточной функции АРФ.

[num2, den2]=lp2bs(num1,den1, Wo, Bw);

% построение АЧХ и ЛАЧХ аналогового РФ:

[H, W] = freqs(num2, den2, 4096);

figure(2)

subplot(2,1,1);

plot (W, abs(H))

xlim([300 400]);

grid;

title('График АЧХ аналогового РФ');

subplot(2,1,2);

plot (W, 20\*log10(abs(H)))

xlim([300 400]);

grid;

title('График ЛАЧХ аналогового РФ');

xlabel('w, рад');

**Приложение 4.** Расчёт передаточной функции ЦРФ.

format long;

[numFil, denFil]=bilinear(num2, den2, Ft);

[H, f]=freqz(numFil, denFil, 4096\*10,Ft);

figure(3)

subplot(2,1,1);

plot(f, abs(H));

xlim([40 60]);

grid;

title('АЧХ цифрового РФ')

subplot(2,1,2), plot(f, 20\*log10(abs(H)))

title('ЛАЧХ цифрового РФ')

xlim([48 52]);

ylim([-70 -50]);

grid;

xlabel('f , Гц')

ylabel('дБ')

figure(4)

zplane(numFil, denFil);

**Приложение 5.** Расчёт коэффициентов каскадной формы.

sos = tf2sos(numFil, denFil) ;

fvtool(sos);

[numc, denc]=sos2tf(sos);

[Hc, fc]=freqz(numc, denc, l, 4096, Ft);

figure(5)

plot(fc,abs(Hc))

hold on;

plot(f, abs(H),'--')

xlim([35 65]);

legend('АЧХ исходного фильтра', 'АЧХ фильтра в каскадной форме')

grid;

**Приложение 6.** Расчёт коэффициентов каскадной формы.

n = 0:300;

y2 =cos(2\*pi\*50\*(1/Ft)\*n); %50 Гц (в полосе задерживания)

y3=cos(2\*pi\*70\*(1/Ft)\*n); %70 Гц (в полосе пропускания)

y1=y2+y3;

n1=[ 0.861888691792640 -0.843859193013008 0.861888693061288];

d1=[1.000000000000000 -0.881800479426981 0.921806264502153];

n2=[1.000000000000000 -1.003983264363036 0.999999998511189];

d2=[1.000000000000000 -1.022392215395558 0.925293847764276];

n3=[1.000000000000000 -0.962613542903116 0.999999999366148];

d3=[1.000000000000000 -0.924328187096459 0.989345165082282];

n4=[1.000000000000000 -1.020011642567023 1.000000000650734];

d4=[1.000000000000000 -1.046054115243146 0.989762247489705];

F1=filter(n1,d1,y1);

F2=filter(n2,d2,F1);

F3=filter(n3,d3,F2);

F4=filter(n4,d4,F3);

figure(5)

subplot(4,1,1), plot(n,y2),

title('Гармоника 50 Гц');

subplot(4,1,2), plot(n,y3),

title('Гармоника 70 Гц');

subplot(4,1,3), plot(n,y1),

title('Сигнал суммы гармоник');

subplot(4,1,4), plot(n,F4),

title('Сигнал на выходе фильтра');

F1=filter(n1,d1,y2);

F2=filter(n2,d2,F1);

F3=filter(n3,d3,F2);

F4=filter(n4,d4,F3);

figure(6)

subplot(2,1,1), plot(n,y2),

title('Гармоника 50 Гц');

subplot(2,1,2), plot(n,F4),

title('Сигнал на выходе фильтра');

20\*log10((y2)/(F4))

F1=filter(n1,d1,y3);

F2=filter(n2,d2,F1);

F3=filter(n3,d3,F2);

F4=filter(n4,d4,F3);

figure(7)

subplot(2,1,1), plot(n,y3),

title('Гармоника 70 Гц');

subplot(2,1,2), plot(n,F4),

title('Сигнал на выходе фильтра');

20\*log10(max(y3)/max(F4))

**Приложение 7.** Вычисления быстродействия фильтра.

#include <stdio.h>

#include <iostream>

#include <math.h>

#include <windows.h>

#include <conio.h>

#pragma warning(disable : 4996)

#define N 100000

# define M\_PI 3.14159265358979323846

using namespace std;

int por = 2, Fd = 300; // порядок фильтров для каскадной формы и частота отсчетов

double b, B, a, A,

y[N], // массив для формирования отсчетов выходного сигнала

x[N], // массив для формирования отсчетов входного сигнала

F1[N], F2[N], F3[N], F4[N]; // массивы передачи сигнала между элементами фильтра

double //коэф-ты полиномов числителя и знаменателя :

n1[3] = { 0.861888691792640, - 0.843859193013008, 0.861888693061288 },

d1[3] = { 1.000000000000000, - 0.881800479426981, 0.921806264502153 },

n2[3] = { 1.000000000000000, - 1.003983264363036, 0.999999998511189 },

d2[3] = { 1.000000000000000, - 1.022392215395558, 0.925293847764276 },

n3[3] = { 1.000000000000000, - 0.962613542903116, 0.999999999366148 },

d3[3] = { 1.000000000000000, - 0.924328187096459, 0.989345165082282 },

n4[3] = { 1.000000000000000, - 1.020011642567023, 1.000000000650734 },

d4[3] = { 1.000000000000000, - 1.046054115243146, 0.989762247489705 },

double Ellips(double x[], int n, double m1[], double m2[])// функция расчета сигнала

{

for (int i = 0; i < por + 1; i++)

{

b = m1[i] \* x[n - i];

B = b + B;

}

for (int i = 1; i < por + 1; i++)

{

a = m2[i] \* y[n - i];

A = a + A;

}

y[n] = B - A;

return y[n];

}

DWORD WINAPI Thread\_filter(LPVOID) // поток для фильтрации сигнала

{

double Ellips(double x[], int n, double m1[], double m2[]);

double count, total;

count = GetTickCount(); // счетчик времени

for (int n = 0; n < N; n++) // фильтрация входного сигнала

{

F1[n] = Ellips(x, n, n1, d1);

F2[n] = Ellips(F1, n, n2, d2);

F3[n] = Ellips(F2, n, n3, d3);

F4[n] = Ellips(F3, n, n4, d4);

}

total = GetTickCount() - count; // вычисление времени фильтрации сигнала

setlocale(LC\_ALL, "Russian");

cout << "Количество отсчётов:" << N << endl;

cout << "Общее время : " << total << " ms" << endl;

cout << "Время на один отсчёт :" << total/N << " ms" << endl;

return 0;

}

iny main()

{

unsigned long ThreadId;

for (int j = 0; j < N; j++)

{

x[j] = 2 \* cos((2 \* M\_PI \* 50 \* j) \* 1 / Fd) + cos((2 \* M\_PI \* 70 \* j) \* 1 / Fd); // входной сигнал

}

CreateThread(NULL, 0, Thread\_filter, x, 0, &ThreadId);//запуск потока фильтрации

return 0;

}