

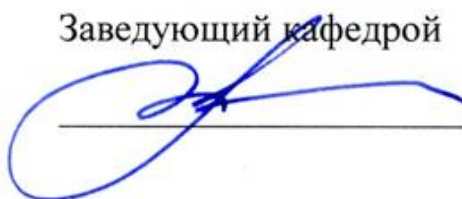
МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Тульский государственный университет»

Институт высокоточных систем им. В.П. Грязева
Кафедра «Системы автоматического управления»

Утверждено на заседании кафедры
«Системы автоматического управления»
«09» декабря 2022 г., протокол № 5

Заведующий кафедрой



О.В.Горячев

МЕТОДИЧЕСКИЕ УКАЗАНИЯ
по выполнению лабораторных работ
по дисциплине(модулю)
«Цифровая фильтрация и цифровая обработка сигналов»

основной профессиональной образовательной программы
высшего образования – программы магистратуры

по направлению подготовки
15.04.06 Мехатроника и робототехника

с направленностью (профилем)
Мехатроника и приводы мехатронных систем

Форма обучения: очная

Идентификационный номер образовательной программы: 150406-03-23

Тула 2022 год

Разработчик методических указаний

Ефромеев А.Г., доц. каф. САУ, к.т.н.


О.Б. Лобачев

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

по выполнению лабораторных работ

по дисциплине «САУ»

«Цифровая обработка сигналов»

основной профессиональной образовательной программы

высшего образования – программы магистратуры

по направлению подготовки

12.04.06 Мехатроника и робототехника

(специальность – Информатика)

Мехатроника и приводы мехатронических систем

Форма обучения: очная

Идентификационный номер образовательной программы: 120406-01-50

Тул 2020 год

Расчёт аналоговых фильтров

ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Аналоговые фильтры

Теория линейных стационарных непрерывных систем широко используется при построении различных частотно-избирательных цепей в электронных модулях мехатронных систем.

Аналоговым фильтрами называется частотно-избирательная цепь, обеспечивающая пропускание сигналов в определенных полосах частот и подавление в других. Область частот, в которой фильтр пропускает сигнал, называют полосой пропускания, а область частот, в которой ослабление входного сигнала велико - полосой задерживания.

Фильтры применяют для выделения требуемого сигнала из смеси полезных и нежелательных сигналов. В зависимости от взаимного расположения полосы пропускания и полосы задерживания различают следующие виды фильтров:

- *фильтры нижних частот* (ФНЧ; английский термин — low-pass filter), пропускающие частоты, меньшие некоторой частоты среза ω_0 ;
- *фильтры верхних частот* (ФВЧ; английский термин — high-pass filter), пропускающие частоты, большие некоторой частоты среза ω_0 ;
- *полосовые фильтры* (ПФ; английский термин — band-pass filter), пропускающие частоты в некотором диапазоне $\omega_1 \dots \omega_2$ (они могут также характеризоваться средней частотой $\omega_0 = (\omega_1 + \omega_2)/2$ и шириной полосы пропускания $\Delta\omega = \omega_2 - \omega_1$);
- *режекторные фильтры* (другие возможные названия — заграждающий фильтр, фильтр-пробка, полосно-задерживающий фильтр; английский термин — band-stop filter), пропускающие на выход все частоты, кроме лежащих в некотором диапазоне $\omega_1 \dots \omega_2$ (они тоже могут характеризоваться средней частотой $\omega_0 = (\omega_1 + \omega_2)/2$ и шириной полосы задерживания $\Delta\omega = \omega_2 - \omega_1$).

Идеальная форма АЧХ фильтров этих четырех типов показана на рис. 1. Однако такая идеальная форма АЧХ не может быть физически реализована. Поэтому в теории аналоговых фильтров разработан ряд методов аппроксимации прямоугольных АЧХ.

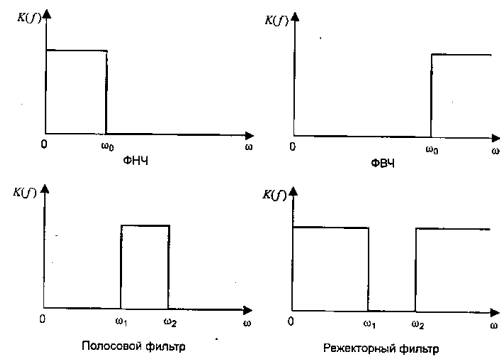


Рис. 1. АЧХ фильтров различного типа.

Фильтр Баттерворта

Передаточная функция фильтра Баттерворта (Butterworth filter)

передаточная функция

не имеет нулей, а ее полюсы равномерно расположены на s -плоскости в левой половине окружности единичного радиуса.

Благодаря такому размещению полюсов формула для АЧХ фильтра Баттерворта оказывается очень простой:

$$K(\omega) = \frac{1}{\sqrt{1 + \left(\frac{\omega}{\omega_0}\right)^{2n}}},$$

где частота среза ω (для фильтра-прототипа она равна 1 рад/с), n - порядок фильтра.

Коэффициент передачи на нулевой частоте равен 1, а на частоте среза независимо от порядка фильтра составляет $1/\sqrt{2} \approx 0.707 \approx 3\text{дБ}$. При АЧХ $\omega \rightarrow \infty$ стремится к нулю.

АЧХ фильтра Баттерворта является максимально плоской при $\omega = 0$ и $\omega \rightarrow \infty$. Это означает, что в данных точках равны нулю производные АЧХ по частоте до $(2n - 1)$ порядка включительно.

В целом АЧХ монотонно спадает от единицы до нуля при изменении частоты от нуля до бесконечности.

Фильтр Чебышева первого рода

Передаточная функция фильтра Чебышева первого рода (Chebyshev type I filter)

Передаточная функция

также не имеет нулей, а ее полюсы расположены в левой половине эллипса на s -плоскости.

АЧХ фильтра Чебышева первого рода описывается следующим образом:

$$K(\omega) = \frac{1}{\sqrt{1 + \varepsilon^2 T_n^2\left(\frac{\omega}{\omega_0}\right)}}$$

Здесь ω - частота среза, $T_n(x)$ - полином Чебышева n -го порядка, n - порядок фильтра, ε - параметр, определяющий величину пульсаций АЧХ в полосе пропускания.

Полином Чебышева $T_n(x)$ при $|x| \leq 1$ колеблется в диапазоне $-1 \dots +1$, а при $|x| > 1$ неограниченно возрастает по абсолютной величине. Поэтому АЧХ фильтра Чебышева первого рода в полосе пропускания (при $|\omega| \leq \omega_0$) колеблется между значениями $\frac{1}{\sqrt{1 + \varepsilon^2}}$ и 1, а вне полосы пропускания (при $|\omega| > \omega_0$) монотонно затухает до нуля.

Коэффициент передачи на нулевой частоте равен 1 при нечетном порядке фильтра и $\frac{1}{\sqrt{1 + \varepsilon^2}}$ при четном. На частоте среза коэффициент передачи фильтра равен $\frac{1}{\sqrt{1 + \varepsilon^2}}$, то есть уровню пульсаций АЧХ в полосе пропускания. При $\omega \rightarrow \infty$ АЧХ стремится к нулю.

По сравнению с фильтром Баттерворта того же порядка фильтр Чебышева обеспечивает более крутой спад АЧХ в области перехода от полосы пропускания к полосе задерживания.

Значение параметра ε и уровень пульсаций R_p (в децибелах) связаны следующим образом:

$$R_p = 20 \lg\left(\sqrt{1 + \varepsilon^2}\right) = 10 \lg(1 + \varepsilon^2) \text{ дБ},$$
$$\varepsilon = \sqrt{10^{R_p/10} - 1}.$$

При $\omega \rightarrow \infty$ АЧХ фильтра Чебышева первого рода является максимально плоской.

Фильтр Чебышева второго рода

Передаточная функция фильтра Чебышева второго рода (Chebyshev type II filter)

Передаточная функция

в отличие от предыдущих случаев, имеет и нули, и полюсы. Она связана с функцией передачи фильтра Чебышева первого рода следующим образом:

$$H_2(s) = 1 - H_1(1/s).$$

Здесь $H_1(s)$ и $H_2(s)$ функции передачи фильтров Чебышева первого и второго рода соответственно.

Полюсы функции передачи фильтров Чебышева первого и второго рода (p_{1i} и p_{2i} соответственно) связаны друг с другом соотношением $p_{2i} = 1/p_{1i}$.

По этой причине фильтры Чебышева второго рода называют еще *инверсными фильтрами Чебышева* (inverse Chebyshev filter).

АЧХ фильтра Чебышева второго рода описывается следующим образом:

$$K(\omega) = \frac{1}{\sqrt{1 + \frac{\varepsilon^2}{T_n^2(\omega/\omega_0)}}}$$

Здесь ω - частота среза, $T_n(x)$ - полином Чебышева n -го порядка, n - порядок фильтра, ε - параметр, определяющий величину пульсаций АЧХ в полосе задерживания.

В результате указанного выше преобразования передаточной функции АЧХ фильтра Чебышева второго рода ведет себя следующим образом: в полосе пропускания она монотонно затухает, а в полосе задерживания колеблется между нулем и значением

$$1/\sqrt{1 + \varepsilon^2}.$$

Коэффициент передачи фильтра на нулевой частоте равен 1, на частоте среза — заданному уровню пульсаций в полосе задерживания. При $\omega \rightarrow \infty$ коэффициент передачи равен нулю при нечетном и уровню пульсаций - при четном порядке фильтра.

Значение параметра ε и уровень пульсации R_s (в децибелах) связаны следующим образом:

$$R_s = 20 \lg(\sqrt{1 + \varepsilon^2}) = 10 \lg(1 + \varepsilon^2) \text{ дБ},$$

$$\varepsilon = \sqrt{10^{R_s/10} - 1}.$$

При $\omega_0=0$ АЧХ фильтра Чебышева второго рода является максимально плоской.

Фильтр Бесселя

В отличие от фильтров предыдущих типов, фильтры Бесселя (Bessel filter) не аппроксимируют прямоугольную АЧХ - их АЧХ по форме близка к гауссовой кривой

(точнее, стремится к ней с ростом порядка фильтра). Практическая ценность фильтров Бесселя определяется тем, что для них зависимость группового времени задержки от частоты является максимально гладкой в точке $\omega=0$ и групповая задержка очень мало меняется в полосе пропускания.

Передаточная функция фильтра Бесселя имеет только полюсы, лежащие на окружности с центром в положительной области вещественной оси. Сама передаточная функция имеет следующий вид:

$$H(s) = \frac{d_0}{\sum_{k=0}^n d_k s^k}.$$

Коэффициенты полинома знаменателя рассчитываются по следующей формуле:

$$d_k = \frac{(2n-k)!}{2^{n-k} k!(n-k)!}.$$

Расчёт аналоговых фильтров в MATLAB.

Для расчета аналогового фильтра необходимо выполнить две основные операции: рассчитать ФНЧ - прототип и преобразовать его к нужному типу фильтра с заданными частотами среза.

Расчёт ФНЧ - прототипа

В MATLAB расчёт аналоговых фильтров-прототипов для рассмотренных фильтров производится с помощью следующих функций:

- $[z,p,k]=\text{butter}(n)$ – для расчёта аналогового фильтра-прототипа Баттерворта;
- $[z,p,k]=\text{cheb1ap}(n,R_p)$ – для расчёта аналогового фильтра-прототипа Чебышева I-го рода;
- $[z,p,k]=\text{cheb2ap}(n,R_s)$ – для расчёта аналогового фильтра-прототипа Чебышева II-го рода;
- $[z,p,k]=\text{besselap}(n)$ – для расчёта аналогового фильтра-прототипа Бесселя.

Входные параметры для соответствующих фильтров-прототипов:

n – порядок фильтра (целочисленный);

R_p – уровень пульсаций в полосе пропускания (в дБ);

R_s – уровень пульсаций в полосе задерживания (в дБ).

Все функции MATLAB для расчета аналоговых прототипов возвращают векторы-столбцы нулей и полюсов функции передачи, а также значение коэффициента усиления.

Преобразование способов описания линейных цепей

Необходимость в преобразовании описаний часто возникает из-за того, что функции расчета цепей (такие как рассматриваемые функции расчета фильтров-прототипов) дают результат в одной форме, а функция, например, построения частотной характеристики, требует задания входных параметров в другой форме.

Формы описания цепей в именах функций обозначаются следующим образом:

tf - коэффициенты полиномов числителя и знаменателя функции передачи (transfer function);

zp - нули и полюсы (zeros and poles);

ss - описание в пространстве состояний (state-space).

Далее кратко рассматриваются конкретные функции преобразования описаний цепей. Для входных и выходных параметров используются следующие обозначения:

- функция передачи:

b - вектор-строка коэффициентов (в порядке убывания степеней) числителя функции передачи;

a - вектор-строка коэффициентов (в порядке убывания степеней) знаменателя функции передачи;

- нули и полюсы:

z - вектор нулей (столбец);

p - вектор полюсов (столбец);

k - коэффициент усиления (скаляр);

- пространство состояний:

A - квадратная матрица связи вектора состояния и его производной;

B - вектор-столбец связи входного сигнала и производной вектора состояния;

C - вектор-строка связи выходного сигнала и вектора состояния;

D - скалярный коэффициент связи выходного и входного сигналов.

Функция *tf2zp* преобразует наборы коэффициентов полиномов числителя и знаменателя функции передачи в векторы нулей и полюсов, рассчитывая также значение общего коэффициента усиления:

$$[z, p, k] = \text{tf2zp}(b, a)$$

Функция *zp2tf* является обратной по отношению к функции *tf2zp*: она осуществляет преобразование коэффициента усиления, а также векторов нулей и полюсов функции передачи в коэффициенты полиномов ее числителя и знаменателя:

$$[b, a] = zp2tf(z, p, k)$$

Функция *tf2ss* преобразует наборы коэффициентов полиномов числителя и знаменателя функции передачи в параметры представления цепи в пространстве состояний:

$$[A, B, C, D] = tf2ss(b, a)$$

Функция *ss2tf* является обратной по отношению к функции *tf2ss*: она преобразует параметры пространства состояний в коэффициенты полиномов функции передачи цепи:

$$[b, a] = ss2tf(A, B, C, D);$$

Функция *zp2ss* преобразует нули, полюсы и коэффициент усиления цепи в ее параметры пространства состояний:

$$[A, B, C, D] = zp2ss(z, p, k)$$

Функция *ss2zp* является обратной по отношению к функции *zp2ss*, преобразуя параметры пространства состояний в нули, полюсы и коэффициент усиления цепи:

$$[z, p, k] = ss2zp(A, B, C, D)$$

Функция *residue*

Идентификатор последней функции, позволяющей преобразовывать описания линейных цепей, выпадает из общего ряда. Это связано с тем, что данное преобразование не является специфическим для обработки сигналов - оно сводится к разложению дробно-рациональной функции на простейшие дроби и часто применяется в математике. По этой же причине данная функция относится не к пакету *Signal Processing*, а к базовой библиотеке *MATLAB*.

Преобразование функции передачи, заданной в виде коэффициентов полиномов числителя и знаменателя, в сумму простых дробей производится с помощью функции *residue*. Она же осуществляет и обратное преобразование; нужное направление преобразования выбирается в зависимости от числа входных параметров.

При двух входных параметрах производится разложение функции передачи на простые дроби:

$$[r, p, k] = residue(b, a)$$

Здесь *b* и *a* — коэффициенты полиномов числителя и знаменателя функции передачи соответственно. Выходные параметры — векторы-столбцы полюсов (*p*) и соответствующих им вычетов (*r*), а также строка коэффициентов целой части *k*.

Преобразования фильтров-прототипов

Следующим этапом после расчета фильтра-прототипа является его преобразование с целью получения фильтра заданного вида с требуемыми частотами среза. Для этого используются приведенные ниже четыре функции MATLAB. Принцип составления их имен следующий: сначала идет сокращение lp, означающее, что исходным фильтром является ФНЧ (low-pass), потом следует символ преобразования 2 и в конце стоит обозначение типа результирующего фильтра:

- lp2lp - изменение частоты среза ФНЧ (low-pass);
- lp2hp - преобразование ФНЧ в ФВЧ (high-pass);
- lp2bp - преобразование ФНЧ в полосовой фильтр (band-pass);
- lp2bs - преобразование ФНЧ в режекторный фильтр (band-stop).

Все эти функции могут преобразовывать фильтры, заданные двумя способами - в виде коэффициентов полиномов числителя и знаменателя функции передачи либо в пространстве состояний. Различаются эти два варианта по числу входных и выходных параметров.

Изменение частоты среза ФНЧ

Изменение частоты среза ФНЧ - прототипа сводится к простому масштабированию частотной оси и выполняется путем следующей замены переменной s в выражении для функции передачи: $s \rightarrow \frac{s}{\omega_0}$, где ω_0 - требуемая частота среза ФНЧ. Такое преобразование производится функцией lp2lp:

$$[bl, al] = lp2lp(b, a, w0)$$

$$[Al, Bl, Cl, D1] = lp2lp(A, B, C, D, w0)$$

Входными параметрами функции являются описание фильтра (в виде коэффициентов полиномов числителя и знаменателя функции передачи - b, a, или в пространстве состояний - A, B, C, D) и требуемая частота среза w0.

Возвращаемый результат - пересчитанные параметры фильтра.

Преобразование ФНЧ в ФВЧ

Преобразование ФНЧ-прототипа в ФВЧ требует инверсии частотной оси и выполняется путем следующей замены переменной s в выражении для функции передачи:

$$s \rightarrow \frac{s}{\omega_0}, \text{ где } \omega_0 - \text{требуемая частота среза ФВЧ.}$$

Такое преобразование производится функцией lp2hp:

$$[bl, al] = lp2hp(b, a, w0)$$

$$[A1, B1, C1, D1] = \text{lp2hp}(A, B, C, D, w0)$$

Входными параметрами функции являются описание фильтра (в виде коэффициентов полиномов числителя и знаменателя функции передачи - b, a или в пространстве состояний - A, B, C, D) и требуемая частота среза w0.

Возвращаемый результат - пересчитанные параметры фильтра.

Преобразование ФНЧ в полосовой фильтр

Преобразование ФНЧ-прототипа в полосовой фильтр требует более сложной трансформации частотной оси, чем в предыдущих случаях. Так, нулевая и бесконечная частоты должны преобразовываться в бесконечное значение на частотной оси ФНЧ-прототипа (там, где его коэффициент передачи стремится к нулю). Частоты, соответствующие краям требуемой полосы пропускания, должны после преобразования давать значения ± 1 , равные частоте среза ФНЧ-прототипа. Наконец, преобразование должно выполняться с помощью дробно-рациональной функции, чтобы сохранить дробно-рациональную структуру функции передачи.

Перечисленным требованиям удовлетворяет следующая замена переменной s:

$$s \rightarrow Q \frac{\left(\frac{s}{\omega_0}\right)^2 + 1}{\frac{s}{\omega_0}},$$

где $\omega_0 = \sqrt{\omega_1 \omega_2}$, $Q = \omega_0 / (\omega_2 - \omega_1)$, ω_1 и ω_2 соответственно нижняя и верхняя границы полосы пропускания фильтра.

Такое преобразование производится функцией lp2bp:

$$[bl, al] = \text{lp2bp}(b, a, w0, Bw)$$

$$[A1, B1, C1, D1] = \text{lp2bp}(A, B, C, D, w0, Bw)$$

Входными параметрами функции являются описание фильтра (в виде коэффициентов полиномов числителя и знаменателя функции передачи - b, a или в пространстве состояний - A, B, C, D), средняя частота w0 и ширина Bw полосы пропускания фильтра (в рад/сек).

Возвращаемый результат - пересчитанные параметры фильтра.

Преобразование ФНЧ в режсекторный фильтр

Для преобразования ФНЧ-прототипа в режсекторный фильтр трансформация частотной оси должна быть обратной по отношению к предыдущему случаю. Нулевая и бесконечная частоты должны преобразовываться в нулевое значение на частотной оси ФНЧ-прототипа (там, где коэффициент передачи велик). Частоты, соответствующие

краям требуемой полосы задерживания, должны после преобразования давать значения ± 1 , равные частоте среза ФНЧ-прототипа. Кроме того, некоторое значение частоты в полосе задерживания должно преобразовываться в бесконечность (там, где коэффициент передачи ФНЧ-прототипа стремится к нулю). Наконец, преобразование должно выполняться с помощью дробно-рациональной функции, чтобы сохранить дробно-рациональную структуру функции передачи.

Перечисленным требованиям удовлетворяет следующая замена переменной s :

$$s \rightarrow \frac{s/\omega_0}{Q \left[\left(s/\omega_0 \right)^2 + 1 \right]},$$

где $\omega_0 = \sqrt{\omega_1 \omega_2}$, $Q = \omega_0 / (\omega_2 - \omega_1)$, ω_1 и ω_2 соответственно нижняя и верхняя границы полосы задерживания фильтра.

Такое преобразование выполняется функцией `1p2bs`:

`[bl, al] = 1p2bs(b, a, w0, Bw)`

`[Al, Bl, Cl, D1] = 1p2bs(A, B, C, D, w0, Bw)`

Входными параметрами функции являются описание фильтра (в виде коэффициентов полиномов числителя и знаменателя функции передачи - b, a или в пространстве состояний - A, B, C, D), средняя частота $w0$ и ширина Bw полосы задерживания фильтра (в радианах в секунду).

Возвращаемый результат - пересчитанные параметры фильтра.

Все сказанное выше применительно к полосовым фильтрам о связи средней частоты с частотами среза и об асимметрии характеристик справедливо и для режекторных фильтров.

Рассмотренная выше требуемая последовательность действий оформлена в виде следующих функций MATLAB:

`butter(n, w0, type, 's')` - расчет фильтров Баттерворта;

`cheby1(n, Rp, w0, type, 's')` - расчет фильтров Чебышева первого рода;

`cheby2(n, Rs, w0, type, 's')` - расчет фильтров Чебышева второго рода;

`besself(n, w0, type)` - расчет фильтров Бесселя.

Параметры всех функций задаются одинаково.

Перечисленные функции, за исключением функции `besself`, позволяют рассчитывать как аналоговые, так и дискретные фильтры. Признаком аналогового расчета служит строка 's', использованная в качестве последнего входного параметра. Параметры `n`, `Rp`, `Rs` (их состав зависит от типа фильтра) — это параметры фильтра-прототипа: `n` — порядок фильтра, `Rp` — уровень пульсаций в полосе пропускания (в децибелах), `Rs` — уровень пульсаций в полосе задерживания (в децибелах).

Параметры `w0` и `type` используются совместно для задания типа фильтра и значений его частот среза (в радианах в секунду):

- ФНЧ: `w0` - скаляр, параметр `type` отсутствует;
- ФВЧ: `w0` - скаляр, `type='high'`;
- полосовой фильтр: `w0` - двухэлементный вектор частот среза [`w1 w2`], параметр `type` отсутствует;
- режекторный фильтр: `w0` - двухэлементный вектор частот среза [`w1 w2`], `type='stop'`.

В зависимости от того, сколько выходных параметров указано при вызове, функции могут возвращать результаты расчета в виде коэффициентов полиномов числителя и знаменателя функции передачи (два выходных параметра), нулей и полюсов (три выходных параметра) либо параметров пространства состояний (четыре выходных параметра):

[`b`, `a`] = ...

[`z`, `p`, `k`] = ...

[`A`, `B`, `C`, `D`] = ...

С учетом всего сказанного перечислим действия, выполняемые функциями расчета аналоговых фильтров:

1. Производится расчет фильтра-прототипа с заданными параметрами АЧХ.
2. Полученные нули и полюсы преобразуются в параметры пространства состояний.
3. Производится преобразование фильтра-прототипа к требуемому типу с заданными частотами среза.
4. Выполняется преобразование описания фильтра к заданному при вызове виду.

Выбор порядка фильтра

Рассмотренные выше функции расчета фильтров требуют задания в качестве входных параметров порядка фильтра и его частоты среза. При этом понятие частоты среза для фильтров разных типов определяется по-разному. Однако исходными данными при

разработке фильтров, как правило, являются другие параметры: частотные границы полос пропускания (ω_p) и задерживания (ω_s), а также допустимая неравномерность АЧХ в полосе пропускания (R_p) и минимально необходимое затухание в полосе задерживания (R_s). Серые области на рисунке демонстрируют допуски, в которые должна укладываться АЧХ фильтра в полосах пропускания и задерживания.

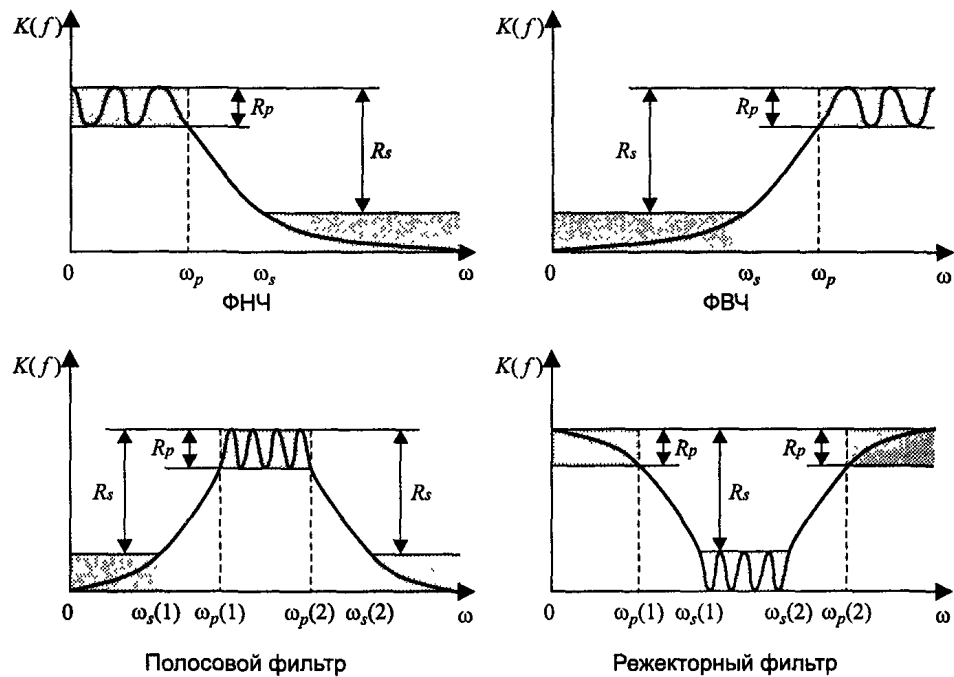


Рис. 2. Задание исходных параметров для расчета ФНЧ, ФВЧ, полосовых и режекторных фильтров.

Выбрать минимально необходимый порядок фильтра позволяют следующие од- нотипные функции пакета Signal Processing:

$$[n, Wn] = \text{buttord}(Wp, Ws, Rp, Rs, 's')$$

$$[n, Wn] = \text{cheblord}(Wp, Ws, Rp, Rs, 's')$$

$$[n, Wn] = \text{cheb2ord}(Wp, Ws, Rp, Rs, 's')$$

Входной параметр R_p - допустимый уровень пульсаций в полосе пропускания (в децибелах), R_s - минимально необходимое затухание в полосе задерживания (в децибелах). Параметры W_p и W_s задают границы полос пропускания и задерживания, способ задания этих параметров зависит от типа проектируемого фильтра:

- ФНЧ: W_p и W_s — числа, при этом должно выполняться неравенство $W_p < W_s$;
- ФВЧ: W_p и W_s — числа, при этом должно выполняться неравенство $W_p > W_s$;
- полосовой фильтр: W_p и W_s — двухэлементные векторы, при этом должны выполняться неравенства $W_s(1) < W_p(1) < W_p(2) < W_s(2)$;

■ режекторный фильтр: W_p и W_s — двухэлементные векторы, при этом должны выполняться неравенства $W_p(1) < W_s(1) < W_s(2) < W_p(2)$.

Значения параметров W_p и W_s были обозначены на рис. 2 как ω_p и ω_s соответственно.

Выходными параметрами являются минимально необходимый для выполнения заданных требований порядок фильтра n и частота среза фильтра W_n . Эти параметры должны затем использоваться при вызове функции расчета фильтра. Возврат значения W_n избавляет пользователя от забот, связанных с тем, что при расчете разных фильтров понятие частоты среза имеет разный смысл.

Поскольку порядок фильтра - величина целочисленная, то обычно оказывается, что фильтр минимально необходимого порядка обеспечивает некоторый запас по исходным параметрам. Этот запас можно использовать по-разному - либо сделать пульсации в полосе пропускания точно равными заданным, но увеличить затухание в полосе задерживания, либо точно выдержать заданное затухание в полосе задерживания, уменьшив при этом пульсации в полосе пропускания. Поведение функций выбора порядка фильтра в этом аспекте определяется тем, что при расчете фильтра должны будут использоваться те же параметры пульсаций R_p и R_s , что и при выборе порядка фильтра. Поэтому для фильтров Баттерворта и Чебышева первого рода будет увеличиваться затухание в полосе задерживания, для фильтров Чебышева второго рода - уменьшаться пульсации в полосе пропускания.

ПРАКТИЧЕСКАЯ ЧАСТЬ

Рассмотрим реализацию аналогового фильтра верхних частот в MATLAB на примере фильтра Баттерворта 3-го порядка с частотой среза $\omega_0=10$ рад/сек.

1. Для расчёта характеристик фильтра-прототипа необходимо создать m-файл. Листинг программы представлен на рис. 3.

```
n=3
[z,p,k]=buttap(n)           % порядок фильтра
plot(p,'x')                 % нули и полюса прототипа
axis equal
axis([-1.5 1.5 -1.5 1.5])
w=0:0.01:5;
[b,a]=zp2tf(z,p,k)          % коэффициенты функции
h=freqs(b,a,w);              % передачи
figure                       % комплексный коэффициент
plot(w,abs(h)), grid        % передачи
figure                       % вызов окна построения
plot(w,unwrap(angle(h))), grid % графика
                               % график АЧХ

                               % график ФЧХ
```

Рис. 3. Листинг программы расчёта фильтра-прототипа
Результаты расчёта представлены на рис. 4.

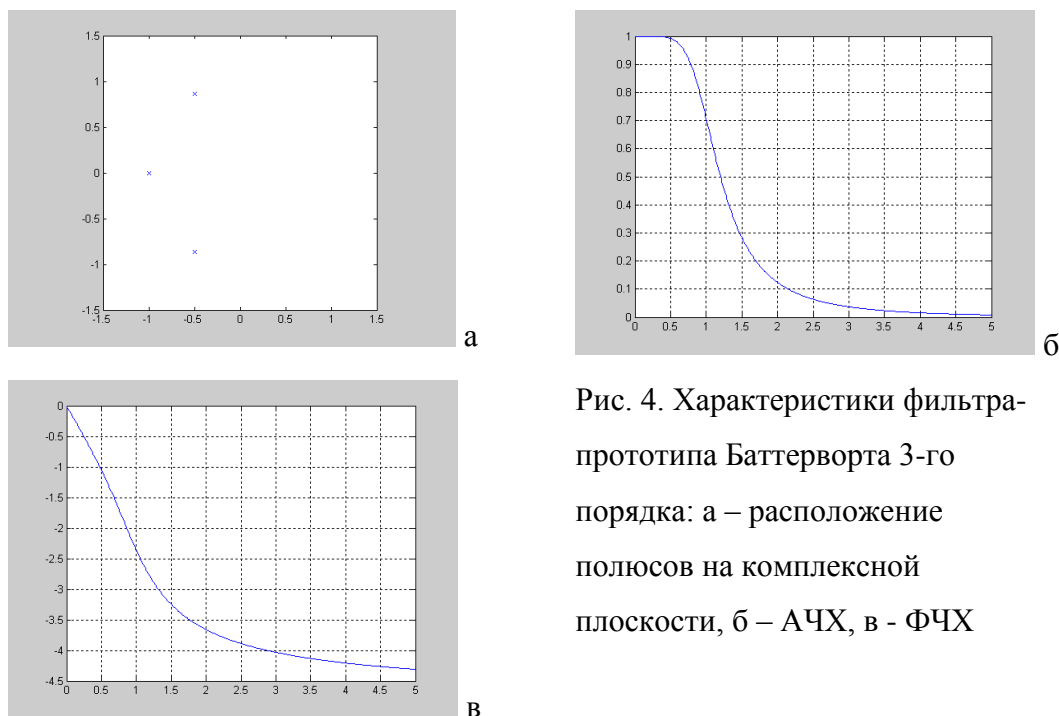


Рис. 4. Характеристики фильтра-прототипа Баттерворта 3-го порядка: а – расположение полюсов на комплексной плоскости, б – АЧХ, в - ФЧХ

2. Преобразуем ФНЧ в ФВЧ с заданной частотой среза. Листинг программы представлен на рис. 5.

```
w0=10
[b1, a1]=lp2hp(b, a, w0)
h=freqs(b1,a1,w);
figure
plot(w,abs(h)), grid
figure
plot(w,unwrap(angle(h))), grid
```

% заданная частота среза фильтра
% преобразование ФНЧ - прототипа к ФВЧ
% комплексный коэффициент передачи
% вызов окна построения графика
% график АЧХ
% график ФЧХ

Рис. 5. Листинг программы преобразования ФНЧ-прототипа в ФВЧ
Результаты представлены на рис. 6.

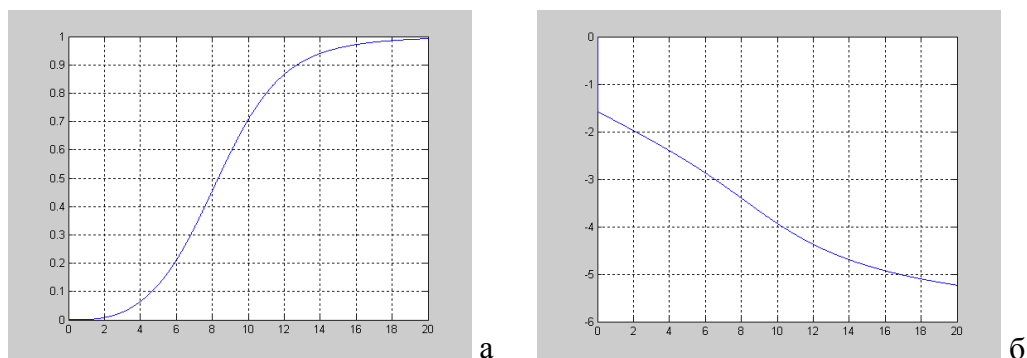


Рис. 6. Характеристики ФВЧ Баттерворта 3-го порядка: а – АЧХ, б - ФЧХ

3. Результаты расчёта фильтра проверяем на модели созданной в SIMULINK. На рис.7. представлена модель системы с рассчитанным фильтром.

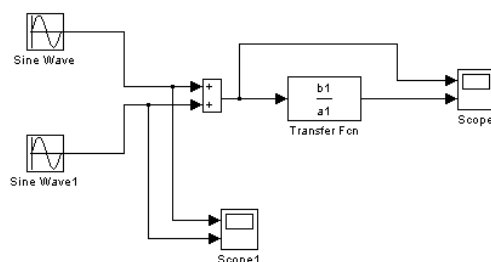


Рис. 7 Модель системы с рассчитанным фильтром.

ЗАДАНИЕ НА РАБОТУ

Изучить аналоговые фильтры.

Изучить основные функции MATLAB, необходимые для расчёта аналоговых фильтров.

Рассчитать аналоговый фильтр.

ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

Изучить аналоговые фильтры.

Изучить основные функции MATLAB, необходимые для расчёта аналоговых фильтров.

В соответствии с планом предложенным в параграфе IV рассчитать аналоговый фильтр с заданными преподавателем параметрами.

Сравнить результаты проведённого расчёта фильтра с результатами расчёта при использовании специальных функций MATLAB непосредственного расчёта фильтров.

Произвести выбор порядка фильтра по заданным преподавателем параметрам.

VI. ОФОРМЛЕНИЕ ОТЧЕТА

Отчёт должен содержать:

- краткие теоретические сведения по выполняемой работе;
- алгоритмы и листинги программ расчёта фильтра;
- выводы по проведённой работе.

VII. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Дайте определения фильтрам НЧ, ВЧ,...
2. Основные характеристики фильтров Баттерворта, Чебышева, Бесселя...
3. Какой из перечисленных типов фильтров имеет наиболее крутой спад АЧХ в области заграждения.
4. Какой из перечисленных типов фильтра имеет минимальный фазовый сдвиг в рабочей полосе частот.
5. Перечислите параметры, которые необходимо задать при расчете фильтров НЧ, ВЧ, режекторных ...

6. Перечислите этапы расчета фильтров в среде MatLab.

VIII. БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Дьяконов В. MATLAB: учебный курс. – СПб: Питер, 2001. – 560 с.: ил.
2. Цифровая обработка сигналов / А.Б. Сергиенко – СПб.: Питер, 2003. – 604 с.: ил.

Спектральный анализ и дискретное преобразование Фурье

I. ЦЕЛЬ И ЗАДАЧИ РАБОТЫ

Изучение спектрального анализа дискретных сигналов. Расчёт спектра дискретного сигнала в Matlab с применением дискретного преобразования Фурье.

II. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Спектр дискретного сигнала

Разработка преобразований Фурье сыграла огромную роль в появлении и развитии ряда новых областей науки и техники. Достаточно отметить, что электрическая связь и радиосвязь базируются на спектральном представлении сигналов. Ряды Фурье также можно рассматривать как приближение произвольных функций (определенные ограничения в этом известны) тригонометрическими рядами бесконечной длины. При конечной длине рядов получаются наилучшие средне-квадратические приближения.

MATLAB содержит функции для выполнения быстрого дискретного преобразования Фурье.

Преобразование Фурье позволяет вычислить спектральную плотность сигнала, представляющего собой *функцию* (как правило, времени либо пространственных координат). Дискретный же сигнал является *последовательностью чисел*, поэтому для анализа его спектра обычными (аналоговыми) средствами необходимо сопоставить этой последовательности некоторую функцию.

Традиционным способом такого сопоставления является представление отсчетов в виде дельта-функций с соответствующими множителями и задержками. Для последовательности отсчетов $\{x(k)\}$ получится следующий сигнал:

$$s(t) = \sum_{k=-\infty}^{\infty} x(k)\delta(t-k). \quad (1)$$

Преобразование Фурье линейно, спектр дельта-функции равен единице, а задержка сигнала во времени приводит к умножению спектра на комплексную экспоненту. Все это позволяет сразу же записать спектр дискретного сигнала:

$$\dot{S}(\omega) = \sum_{k=-\infty}^{\infty} x(k)e^{-j\omega k}. \quad (2)$$

Из этой формулы видно главное свойство спектра любого дискретного сигнала: спектр является периодическим, и его период в данном случае равен 2π (то есть круговой частоте дискретизации, поскольку, составляя сигнал из дельта-функций, мы выбрали единичный интервал между ними, что дает $\omega_D = 2\pi$).

Размерность спектральной функции дискретного сигнала совпадает с размерностью отсчетов. Это связано с тем, что дельта-функции времени, из которых был составлен сигнал (1), имеют размерность частоты.

Формула (2) позволяет вычислить спектральную функцию по известным отсчетам $x(k)$. При конечном числе ненулевых отсчетов этот расчет несложен; он может быть выполнен с помощью функции MATLAB freqz.

Теперь рассмотрим несколько иную задачу. Пусть значения $x(k)$ являются отсчетами аналогового сигнала $s(t)$ взятыми с периодом T :

$$x(k)=s(kT).$$

Выясним, как в этом случае спектр дискретного сигнала (2) связан со спектром аналогового сигнала $s(t)$.

Итак, мы рассматриваем *дискретизированный* сигнал в виде последовательности дельта-функций, «взвешенной» значениями отсчетов $s(kT)$ аналогового сигнала $s(t)$ (рис. 1):

$$s_A(t) = \sum_{k=-\infty}^{\infty} s(kT)\delta(t - kT). \quad (3)$$

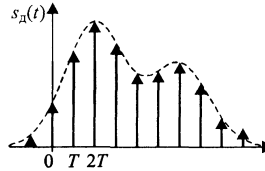


Рис. 1. Дискретизированный сигнал в виде последовательности дельта-функций

Так как функция $\delta(t - kT)$ равна нулю всюду, кроме момента $t = kT$, можно заменить в выражении (3) константы $s(kT)$ на исходный непрерывный сигнал $s(t)$:

$$s_A(t) = s(t) \sum_{k=-\infty}^{\infty} \delta(t - kT). \quad (4)$$

С учётом того, что в интервал интегрирования $(-T/2, T/2)$ попадает только одна дельта-функция, соответствующая $k = 0$, периодическая последовательность дельта-функций может быть представлена в виде комплексного ряда Фурье:

$$\sum_{k=-\infty}^{\infty} \delta(t - kT) = \frac{1}{T} \sum_{n=-\infty}^{\infty} e^{j\omega_n t}, \quad (5)$$

где $\omega_n = 2\pi n/T$. Подставив (5) в (4), получим $s_A(t) = \frac{s(t)}{T} \sum_{n=-\infty}^{\infty} e^{j\omega_n t} = \frac{1}{T} \sum_{n=-\infty}^{\infty} s(t) e^{j\omega_n t}$.

Умножение сигнала на $\exp(j\omega_n t)$ соответствует сдвигу спектральной функции на ω_n , поэтому спектр дискретизированного сигнала можно записать следующим образом:

$$\dot{s}_A(\omega) = \frac{1}{T} \sum_{n=-\infty}^{\infty} \dot{s}\left(\omega - \frac{2\pi n}{T}\right). \quad (6)$$

Таким образом, спектр дискретизированного сигнала представляет собой бесконечный ряд сдвинутых копий спектра исходного непрерывного сигнала $s(t)$ (рис. 2). Расстояние по частоте между соседними копиями спектра равно частоте дискретизации $\omega_d = 2\pi/T$.

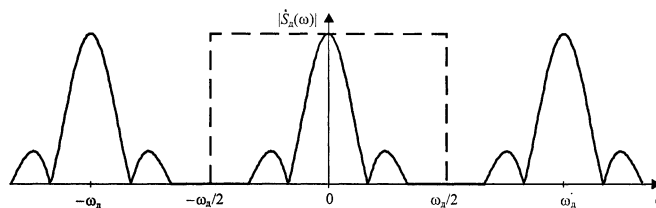


Рис. 2. Спектр дискретизированного сигнала

Характер спектра дискретизированного сигнала еще раз демонстрирует частотно-временную дуальность преобразования Фурье:

периодический сигнал \rightarrow дискретный спектр;

периодический спектр \rightarrow дискретный сигнал.

Рисунок 2 наглядно демонстрирует способ восстановления непрерывного сигнала по дискретным отсчетам. Для этого необходимо пропустить дискретный сигнал через идеальный фильтр нижних частот (ФНЧ) с частотой среза, равной половине частоты дискретизации. АЧХ такого фильтра показана на рис. 2 пунктиром.

Очевидно, что точное восстановление сигнала возможно, если сдвинутые копии спектра не перекрываются. Из рис. 2 видно, что для этого необходимо, чтобы частота дискретизации как минимум в два раза превышала верхнюю граничную частоту в спектре сигнала:

$$\omega_d > 2\omega_A \quad (7)$$

В случае произвольного сигнала, если условие (7) не выполняется, сдвинутые копии спектра будут накладываться друг на друга, что приведет к неизбежным искажениям при восстановлении непрерывного сигнала (рис. 3).

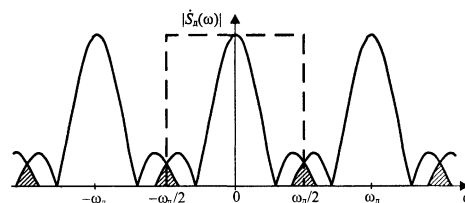
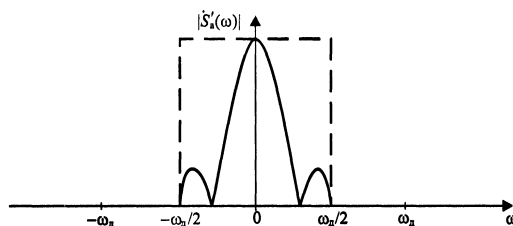
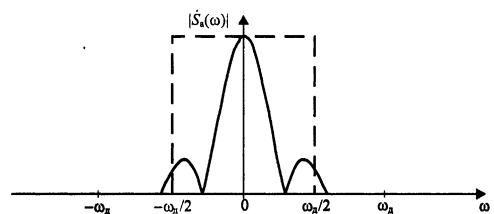


Рис. 3. Перекрывание сдвинутых копий спектра при недостаточно высокой частоте дискретизации

Эти искажения вызваны тем, что спектральные составляющие сигнала с частотами, превышающими частоту Найквиста, равную $\omega_d/2$, не могут быть восстановлены правильно - вместо этого они вызывают наложение «хвостов» соседних сдвинутых копий спектра и появление ложных частот.

Если подлежащий дискретизации сигнал может содержать спектральные составляющие с частотами, превышающими частоту Найквиста, полезно предварительно пропустить его через ФНЧ с частотой среза, равной частоте Найквиста (рис. 4). При этом все равно будут потеряны высокочастотные составляющие — сохранить их можно лишь путем повышения частоты дискретизации. Однако в этом случае благодаря отсутствию наложения «хвостов» не произойдет появления ложных частот и диапазон частот $0 \dots \omega_d/2$ будет представлен в дискретном сигнале без искажений.



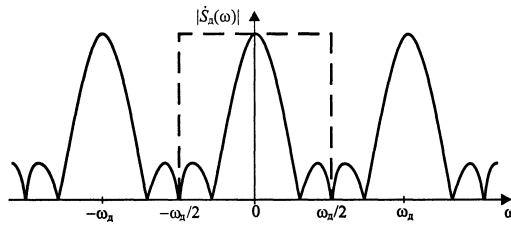


Рис. 4. При дискретизации сигнала, содержащего высокочастотные составляющие (а), полезно пропустить его через ФНЧ (б), чтобы избежать появления ложных частот (в)

Теорема Котельникова

Выше было сформулировано утверждение о том, что некоторые сигналы могут быть без потерь информации представлены своими дискретными отсчетами, а полученное выражение для спектра дискретизированного сигнала позволяет выделить тот класс сигналов, для которых это возможно, и описать способ такого восстановления.

Согласно (6), спектр дискретизированного сигнала представляет собой сумму сдвинутых копий спектра исходного сигнала, при этом шаг сдвига равен частоте дискретизации ω_d . Из рис. 1 видно, что если в спектре аналогового сигнала не содержится составляющих с частотами, превышающими частоту Найквиста ($\omega_d/2$), то сдвинутые копии спектра не будут перекрываться. В этом случае использование идеального ФНЧ с прямоугольной АЧХ позволит выделить исходную (несдвинутую) копию спектра, сосредоточенную в окрестностях нулевой частоты, и, таким образом, в точности восстановить исходный аналоговый сигнал.

АЧХ идеального ФНЧ, восстанавливающего аналоговый сигнал, приведена на рис. 5 слева. Коэффициент передачи в полосе пропускания равен T , а не единице, чтобы компенсировать множитель $1/T$ в формуле (6). С помощью обратного преобразования Фурье найдем импульсную характеристику фильтра (рис. 5 справа):

$$h(t) = \frac{\sin\left(\pi \frac{t}{T}\right)}{\pi \frac{t}{T}}. \quad (8)$$

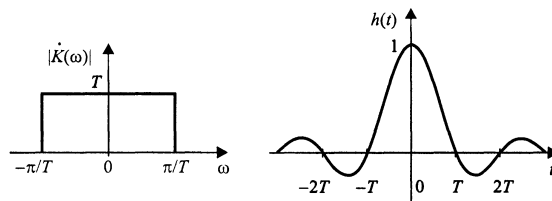


Рис. 5. Амплитудно-частотная (слева) и импульсная (справа) характеристики идеального восстанавливающего фильтра

Дискретизированный сигнал (3) представляет собой сумму дельта-функций. При прохождении такого сигнала через восстанавливающий ФНЧ каждая дельта-функция породит на выходе соответствующим образом сдвинутую и масштабированную копию импульсной характеристики фильтра. Выходной сигнал (в точности соответствующий исходному аналоговому сигналу), таким образом, будет представлять собой сумму сдвинутых и умноженных на отсчеты сигнала копий импульсных характеристик идеального ФНЧ (8):

$$s(t) = \sum_{k=-\infty}^{\infty} s(kT) \frac{\sin\left(\pi \frac{t-kT}{T}\right)}{\pi \frac{t-kT}{T}}. \quad (9)$$

Подводя итог всему сказанному, сформулируем *теорему Котельникова*: любой сигнал $s(t)$, спектр которого не содержит составляющих с частотами выше некоторого значения $\omega_A = 2\pi f_A$, может быть без потерь информации представлен своими дискретными отсчетами $\{s(kT)\}$, взятыми с интервалом T , удовлетворяющим следующему неравенству:

$$T \leq \frac{1}{2f_A} = \frac{\pi}{\omega_A}. \quad (10)$$

Восстановление исходного непрерывного сигнала $s(t)$ по набору его дискретных отсчетов $\{s(kT)\}$ производится по формуле (9).

Формирование непрерывного сигнала по его дискретным отсчетам поясняет рис. 6. Пунктиром показаны графики отдельных слагаемых формулы (9), сплошной линией - восстановленный сигнал. Ниже приводится код MATLAB, использованный при построении рисунка:

```
t = -2:0.01:6; % время для восстановленного сигнала
td = -2:6; % номера отсчетов
s = [0 0 4 3 2 1 0 0 0]; % дискретный сигнал
d = [td' s']; % данные для функции pulstran
y = pulstran(t, d, 'sinc'); % восстановленный сигнал
plot(td, s, 'o', t, y) % график восстановленного сигнала
hold on % выводим графики отдельных sinc-импульсов
for k=1:length(s), plot(t, s(k)*sinc(t-td(k)), ':'), end
hold off
```

Рисунок 6 наглядно демонстрирует главное свойство сигнала с ограниченным спектром - его бесконечность во времени. Хотя отличны от нуля лишь несколько отсчетов показанного сигнала, аналоговый сигнал оказывается бесконечно колеблющимся - между нулевыми отсчетами (на рисунке это отсчеты с номерами -2, -1, 4, 5, 6) его значения отличны от нуля. Эти колебания нигде не заканчиваются, хотя их амплитуда стремится к нулю.

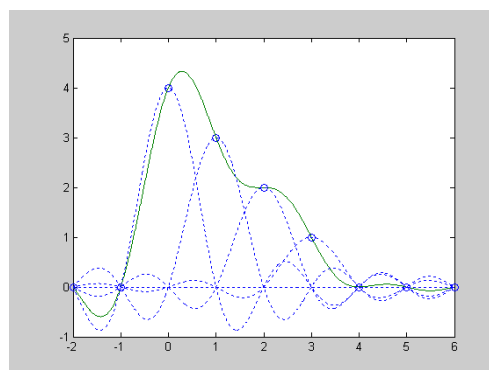


Рис. 6. Восстановление непрерывного сигнала по его дискретным отсчетам

Иногда можно встретить примерно следующее «объяснение» сущности теоремы Котельникова: «если брать отсчеты достаточно часто, в промежутках между ними сигнал с ограниченным спектром не успеет сильно измениться, и мы сможем точно восстановить его». Такая трактовка является принципиально неправильной. Когда мы говорим об ограниченной полосе частот сигнала, имеется в виду спектральная функция *всего сигнала, имеющего бесконечную длительность*. При этом *мгновенные* спектры отдельных фрагментов сигнала могут содержать сколь угодно высокие частоты. В частности, в отдельном промежутке между соседними отсчетами сигнал с ограниченным спектром может иметь сколь угодно сложную форму, например произвольное число раз менять знак.

Дискретное преобразование Фурье

Прямое преобразование Фурье переводит описание сигнала (функции времени) из временной области в частотную, а обратное преобразование Фурье — из частотной области во временную. На этом основаны многочисленные методы фильтрации сигналов.

В разделе «Спектр дискретного сигнала» мы проанализировали явления, происходящие со спектром при дискретизации сигнала. Рассмотрим теперь, что представляет собой спектр дискретного *периодического* сигнала.

Итак, пусть последовательность отсчетов $\{x(k)\}$ является периодической с периодом N :

$$x(k + N) = x(k) \text{ для любого } k.$$

Такая последовательность полностью описывается *конечным* набором чисел, в качестве которого можно взять произвольный фрагмент длиной N , например $\{x(k), k = 0, 1, \dots, N - 1\}$. Поставленный в соответствие этой последовательности сигнал из смещенных по времени дельта-функций:

$$s(t) = \sum_{k=-\infty}^{\infty} x(k)\delta(t - kT) \quad (11)$$

также, разумеется, будет периодическим с минимальным периодом $N T$.

Так как сигнал (11) является дискретным, его спектр должен быть *периодическим* с периодом $2\pi/T$. Так как этот сигнал является также и периодическим, его спектр должен быть *дискретным* с расстоянием между гармониками, равным $2\pi/(NT)$.

Итак, периодический дискретный сигнал имеет периодический дискретный спектр, который также описывается конечным набором из N чисел (один период спектра содержит $\frac{2\pi}{T} / \frac{2\pi}{NT} = N$ гармоник).

Рассмотрим процедуру вычисления спектра периодического дискретного сигнала. Так как сигнал периодический, будем раскладывать его в *ряд Фурье*. Коэффициенты $\dot{x}(n)$ этого ряда равны

$$\dot{x}(n) = \frac{1}{NT} \int_0^{NT} s(t) e^{-j\omega_n t} dt = \frac{1}{NT} \int_0^{NT} \sum_{k=0}^{N-1} x(k)\delta(t - Tk) e^{-j\omega_n t} dt = \frac{1}{NT} \sum_{k=0}^{N-1} x(k) \int_0^{NT} \delta(t - Tk) e^{-j\omega_n t} dt = \frac{1}{NT} \sum_{k=0}^{N-1} x(k) e^{-j\omega_n kT} = \frac{1}{NT} \sum_{k=0}^{N-1} x(k) \exp\left(-j \frac{2\pi nk}{N}\right). \quad (12)$$

Таким образом, формула для вычисления комплексных амплитуд гармоник представляет собой линейную комбинацию отсчетов сигнала.

В выражении (12) реальный масштаб времени фигурирует только в множителе $1/T$ перед оператором суммирования. При рассмотрении дискретных последовательностей обычно оперируют номерами отсчетов и спектральных гармоник без привязки к действительному масштабу времени и частоты. Поэтому множитель $1/T$ из (12) удаляют, то есть считают частоту дискретизации равной единице. Удаляют обычно и множитель

$1/N$. Получившееся выражение называется *дискретным преобразованием Фурье* (ДПФ; английский термин — Discrete Fourier Transform, DFT):

$$\dot{X}(n) = \sum_{k=0}^{N-1} x(k) \exp\left(-j \frac{2\pi nk}{N}\right). \quad (13)$$

Существует и *обратное* дискретное преобразование Фурье. Переход от дискретного спектра к временным отсчетам сигнала выражается следующей формулой:

$$x(k) = \frac{1}{N} \sum_{n=0}^{N-1} \dot{X}(n) \exp\left(j \frac{2\pi nk}{N}\right). \quad (14)$$

Восстановление непрерывного сигнала с помощью ДПФ

Являясь по своей сути спектром *дискретного* периодического сигнала, дискретное преобразование Фурье позволяет легко восстановить *непрерывный* периодический сигнал, занимающий ограниченную полосу частот. Для этого в формуле обратного ДПФ (14) необходимо заменить дискретный параметр (номер отсчета k) на непрерывный — нормированное время t/T , где T — период дискретизации:

$$x(k) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} \dot{X}(n) \exp\left(j \frac{2\pi nt}{NT}\right). \quad (15)$$

Следует обратить внимание на еще одно отличие этого соотношения от формулы (14): диапазон индексов суммирования смещен вниз на $N/2$ (при четном N ; при нечетном N суммирование производится от $n=-(N-1)/2$ до $(N-1)/2$). Это необходимо, чтобы получить аналоговый сигнал, занимающий полосу частот от 0 до π/T . Коэффициенты $\dot{X}(n)$ с отрицательными номерами могут быть получены из соотношения симметрии.

Результат восстановления непрерывного периодического сигнала с помощью ДПФ, разумеется, совпадает с результатами, получаемыми при использовании ряда Котельникова. Однако использование ДПФ в данном случае оказывается более предпочтительным, так как ряд Котельникова для периодического сигнала содержит бесконечное число слагаемых, а формула (15) — конечное.

Матрица ДПФ

ДПФ является линейным преобразованием, трансформирующим вектор временных отсчетов в вектор такой же длины, содержащий отсчеты спектральные. Такое преобразование может быть реализовано как умножение некоторой квадратной матрицы на входной вектор-столбец:

$$\mathbf{y} = \mathbf{A} \mathbf{x}$$

где \mathbf{A} — *матрица преобразования*. В случае ДПФ эта матрица имеет вид

$$A_{DFT} = \begin{bmatrix} 1 & 1 & 1 & 1 & \dots & 1 \\ 1 & e^{-j\frac{2\pi}{N}} & e^{-j\frac{4\pi}{N}} & e^{-j\frac{6\pi}{N}} & \dots & e^{-j\frac{2\pi}{N}(N-1)} \\ 1 & e^{-j\frac{4\pi}{N}} & e^{-j\frac{8\pi}{N}} & e^{-j\frac{12\pi}{N}} & \dots & e^{-j\frac{2\pi}{N}2(N-1)} \\ 1 & e^{-j\frac{6\pi}{N}} & e^{-j\frac{12\pi}{N}} & e^{-j\frac{18\pi}{N}} & \dots & e^{-j\frac{2\pi}{N}3(N-1)} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & e^{-j\frac{2\pi}{N}(N-1)} & e^{-j\frac{2\pi}{N}2(N-1)} & e^{-j\frac{2\pi}{N}3(N-1)} & \dots & e^{-j\frac{2\pi}{N}(N-1)^2} \end{bmatrix}.$$

Общая формула для элемента матрицы, расположенного в n -м столбце m -й строки, выглядит так:

$$A_{DFT}(m, n) = \exp\left(-j2\pi \frac{(m-1)(n-1)}{N}\right), 1 \leq m \leq N, 1 \leq n \leq N.$$

Вычисление ДПФ путем умножения матрицы на вектор полностью соответствует формуле (13). Этот метод требует большого количества вычислительных операций, поэтому на практике вместо него применяются быстрые алгоритмы, рассматриваемые далее.

Связь ДПФ и спектра дискретного сигнала

Имея один и тот же конечный набор чисел, можно рассчитать либо *спектральную функцию* этого дискретного сигнала по формуле (2), либо его ДПФ по формуле (13). Разумеется, возникает вопрос о том, как связаны друг с другом эти два спектральных представления, полученные на основе одних и тех же отсчетов сигнала.

Сравнение формул (2) и (13) показывает, что ДПФ представляет собой просто *дискретные отсчеты* спектральной функции дискретного сигнала, соответствующие частотам $\omega_n = \omega_A n / N$:

$$\dot{X}(n) = \dot{S}\left(\frac{2\pi n}{NT}\right) = \dot{S}\left(\omega_A \frac{n}{N}\right). \quad (16)$$

По этой причине значения ДПФ иногда называют *спектральными отсчетами*.

Из соотношения (16) следует еще один важный вывод: если добавить к конечному набору отсчетов некоторое количество нулей, спектральная функция дискретного сигнала, естественно, не изменится, но ДПФ даст большее число спектральных отсчетов, соответствующих частотам, более тесно расположенным в интервале от нуля до частоты дискретизации.

Поясним сказанное на простом примере, вычислив ДПФ для отсчетов прямоугольного импульса при разном количестве концевых нулей (рис. 7):

```
x1 = [ones(8,1); zeros(8,1)];           % 16 отсчетов

y1 = fft(x1);                           % ДПФ сигнала x1

x2 = [x1; zeros(16,1)];                 % добавляем 16 нулей

stem((0:31)/32, abs(y2))                % модуль ДПФ сигнала x2

subplot(2, 2, 1), stem(0:15, x1)        % график сигнала x1

xlim([0 31])

subplot(2, 2, 2)

stem((0:15)/16, abs(y1))                % модуль ДПФ сигнала x1

subplot(2, 2, 3), stem(0:31, x2)        % график сигнала x2

xlim([0 31])

subplot(2, 2, 4)

y2 = fft(x2);                           % ДПФ сигнала x2
```

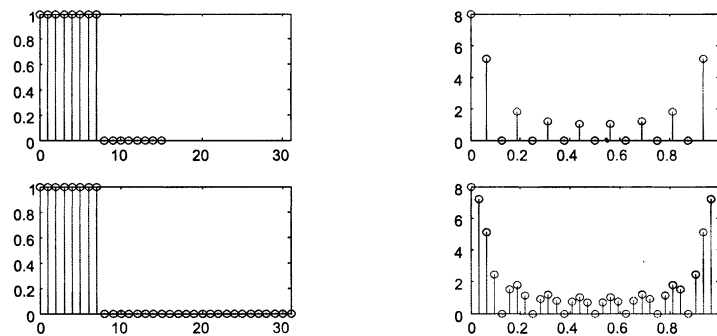


Рис. 7. Повышение спектрального разрешения ДПФ при дополнении сигнала нулями: сверху - исходный сигнал и модуль его ДПФ, снизу - сигнал, дополненный 16 нулями, и модуль его ДПФ

Горизонтальные оси на спектральных графиках проградуированы в долях частоты дискретизации. Из рис. 5.1 видно, что после увеличения длины сигнала вдвое за счет добавления нулевых отсчетов результат ДПФ стал содержать вдвое больше значений, соответствующих в два раза чаще расположенным частотам. Таким образом, дополнение сигнала нулями позволяет повысить спектральное разрешение при вычислении ДПФ.

Функции спектрального анализа в MATLAB

Функция одномерного прямого преобразования Фурье

В описанных ниже функциях реализован особый метод *быстрого преобразования Фурье* — Fast Fourier Transform (FFT, или БПФ), позволяющий резко уменьшить число арифметических операций в ходе приведенных выше преобразований.

Для одномерного преобразования используется следующая функция:

- ✓ `fft(X)` возвращает для вектора X дискретное преобразование Фурье, по возможности используя алгоритм быстрого преобразования Фурье. Если X — матрица, функция `fft` возвращает преобразование Фурье для каждого столбца матрицы;
- ✓ `fft(X,n)` возвращает n -точечное преобразование Фурье. Если длина вектора X меньше n , то недостающие элементы заполняются нулями. Если длина X больше n , то лишние элементы удаляются. Когда X — матрица, длина столбцов корректируется аналогично;
- ✓ `fft(X,[],dim)` и `fft(X,n,dim)` применяют преобразование Фурье к одной из размерностей массива в зависимости от значения параметра `dim`

Функции обратного преобразования Фурье

В одномерном случае возможно обратное преобразование Фурье, реализуемое следующей функцией:

- ✓ `ifft(F)` возвращает результат дискретного обратного преобразования Фурье вектора F . Если F - матрица, то `ifft` возвращает обратное преобразование Фурье для каждого столбца этой матрицы;
- ✓ `ifft(F,n)` возвращает результат n -точечного дискретного обратного преобразования Фурье вектора F ;
- ✓ `ifft(F,[],dim)` и `y = ifft(X,n,dim)` возвращают результат обратного дискретного преобразования Фурье массива F по строкам или по столбцам в зависимости от значения скаляра `dim`.

Для любого X результат последовательного выполнения прямого и обратного преобразований Фурье $\text{ifft}(\text{fft}(x))$ равен X с точностью до погрешности округления. Если X — вещественный массив, $\text{ifft}(\text{fft}(x))$ может иметь малые мнимые части.

III. ПРАКТИЧЕСКАЯ ЧАСТЬ

Для иллюстрации применения преобразования Фурье создадим трехчастотный сигнал на фоне сильного шума, создаваемого генератором случайных чисел:

```
t=0:0.0005:1;  
x=sin(2*pi*200*t)+0.6*sin(2*pi*150*t)+0.8*sin(2*pi*250*t);  
y=x+2*randn(size(t));  
plot(y(1:100))
```

Этот сигнал имеет среднюю частоту 200 Гц и два боковых сигнала с частотами 150 и 250 Гц, что соответствует амплитудно-модулированному сигналу с частотой модуляции 50 Гц и глубиной модуляции 0,8 (амплитуда боковых частот составляет 0,4 от амплитуды центрального сигнала). На рис. 8 показан график этого сигнала (по первым 100 отсчетам из 2000). Нетрудно заметить, что из него никоим образом не видно, что полезный сигнал — амплитудно-модулированное колебание, настолько оно забито шумами.

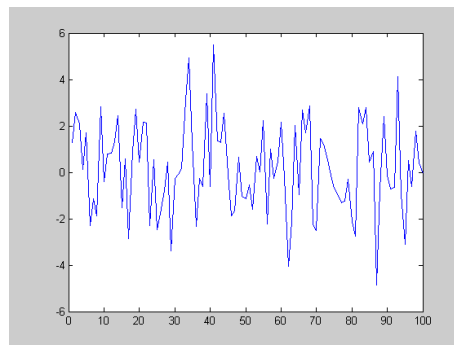


Рис.8 Форма зашумленного сигнала

Теперь построим график спектральной плотности полученного сигнала с помощью прямого преобразования Фурье, по существу переводящего временное представление сигнала в частотное. Этот график (см. рис. 8) в области частот до 300 Гц строится следующими командами:

```
Y=fft(y,1024);  
f=2000*(0:150)/1024;  
figure;  
plot(f,abs(Y(1:151))); grid
```

График спектральной плотности сигнала, построенный в этом примере, представлен на рис. 9.

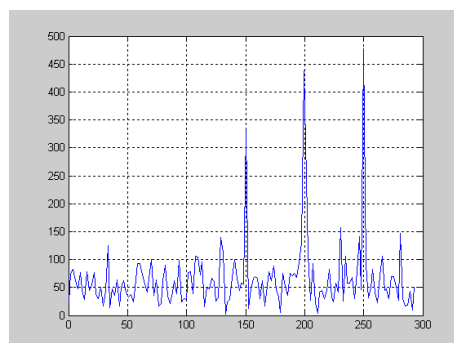


Рис 9. График спектральной плотности приведенного на рис. 8 сигнала

Даже беглого взгляда на рис. 9 достаточно, чтобы убедиться в том, что спектрограмма сигнала имеет явный пик на средней частоте амплитудно-модулированного сигнала и два боковых пика. Все эти три частотные составляющие сигнала явно выделяются на общем шумовом фоне. Таким образом, данный пример наглядно иллюстрирует технику обнаружения слабых сигналов на фоне шумов лежащую в основе работы радиоприемных устройств.

IV. ЗАДАНИЕ НА РАБОТУ

Изучить основные положения теории спектрального анализа дискретных сигналов.

Изучить функции MATLAB дискретного преобразования Фурье, необходимые для спектрального анализа дискретных сигналов.

Рассчитать спектр сигнала.

V. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Изучить основные положения теории спектрального анализа дискретных сигналов.

2. Изучить основные положения теории дискретного преобразования Фурье.

Изучить основные функции MATLAB, необходимые для спектрального анализа дискретных сигналов.

3. В соответствии с планом предложенным в параграфе IV рассчитать спектр сигнала с заданными преподавателем параметрами.

VI. ОФОРМЛЕНИЕ ОТЧЕТА

Отчёт должен содержать:

- краткие теоретические сведения по выполняемой работе;
- алгоритмы и листинги программ расчёта спектра;
- выводы по проведённой работе.

VII. КОНТРОЛЬНЫЕ ВОПРОСЫ

VIII. БИБЛИОГРАФИЧЕСКИЙ СПИСОК

3. Конспект лекций по курсу.
4. Дьяконов В. MATLAB: учебный курс. – СПб: Питер, 2001. – 560 с.: ил.
5. Цифровая обработка сигналов / А.Б. Сергиенко – СПб.: Питер, 2003. – 604 с.: ил.

Спектральный анализ непрерывных и дискретных сигналов. Быстрое преобразование Фурье.

I. ЦЕЛЬ И ЗАДАЧИ РАБОТЫ

Изучение алгоритма быстрого преобразования Фурье дискретных сигналов.
Вычисление 8-точечного ДПФ с помощью алгоритма БПФ с прореживанием по времени.
Проверка полученных вычислений в Matlab.

VI. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Алгоритм быстрого преобразования Фурье

Для вычисления одного коэффициента ДПФ по формуле (13) (см. л.р. №2) необходимо выполнить N комплексных умножений и сложений. Таким образом, расчет всего ДПФ, содержащего N коэффициентов, потребует N^2 пар операций «умножение-сложение». Число операций возрастает пропорционально квадрату размерности ДПФ. Однако, если N не является простым числом и может быть разложено на множители, процесс вычислений можно ускорить, разделив анализируемый набор отсчетов на части, вычислив их ДПФ и объединив результаты. Такие способы вычисления ДПФ называются *быстрым преобразованием Фурье* (БПФ; английский термин — Fast Fourier Transform, FFT) и повсеместно используются на практике.

При реализации БПФ возможно несколько вариантов организации вычислений в зависимости от способа деления последовательности отсчетов на части (*прореживание по времени* либо по частоте) и от того, на сколько фрагментов производится разбиение последовательности на каждом шаге (*основание БПФ*).

БПФ с прореживанием по времени

Рассмотрим идею БПФ с *прореживанием по времени* (decimation in time, DIT) на примере деления набора отсчетов пополам.

Итак, пусть N - четное число. Выделим в формуле (13) два слагаемых, соответствующих элементам исходной последовательности с четными и нечетными номерами:

$$\dot{X}(n) = \sum_{m=0}^{N/2-1} x(2m)e^{-j\frac{2\pi 2mn}{N}} + \sum_{m=0}^{N/2-1} x(2m+1)e^{-j\frac{2\pi(2m+1)n}{N}}.$$

Введем обозначения $y(m) = x(2m)$ и $z(m) = x(2m+1)$, а также вынесем из второй суммы общий множитель $e^{-j\frac{2\pi mn}{N}}$:

$$\dot{X}(n) = \sum_{m=0}^{N/2-1} y(m)e^{-j\frac{2\pi mn}{N/2}} + e^{-j\frac{2\pi n}{N}} \sum_{m=0}^{N/2-1} z(m)e^{-j\frac{2\pi mn}{N/2}}. \quad (17)$$

Две суммы в (17) представляют собой ДПФ последовательностей $\{y(m)\}$ (отсчеты с четными номерами) и $\{z(m)\}$ (отсчеты с нечетными номерами). Каждое из этих ДПФ имеет размерность $N/2$. Таким образом,

$$\dot{X}(n) = \dot{Y}(n) + e^{-j\frac{2\pi n}{N}} \dot{Z}(n), \quad (18)$$

где $Y(n)$ и $Z(n)$ - ДПФ соответственно последовательностей отсчетов с четными и нечетными номерами:

$$\begin{aligned} \dot{Y}(n) &= \sum_{m=0}^{N/2-1} y(m)e^{-j\frac{2\pi mn}{N/2}}, \\ \dot{Z}(n) &= \sum_{m=0}^{N/2-1} z(m)e^{-j\frac{2\pi mn}{N/2}}. \end{aligned}$$

Так как ДПФ размерности $N/2$ дает лишь $N/2$ спектральных коэффициентов, непосредственно использовать формулу (18) можно только при $0 \leq n < N/2$. Для остальных n ($N/2 \leq n < N$) следует воспользоваться периодичностью спектра дискретного сигнала (и, соответственно, периодичностью результатов ДПФ):

$$\dot{Y}\left(n + \frac{N}{2}\right) = \dot{Y}(n), \dot{Z}\left(n + \frac{N}{2}\right) = \dot{Z}(n).$$

С учетом этого при $n \geq N/2$ формула (18) представляется в виде

$$\dot{X}(n) = \dot{Y}\left(n - \frac{N}{2}\right) + e^{-j\frac{2\pi}{N}\left(n - \frac{N}{2}\right)} \dot{Z}\left(n - \frac{N}{2}\right) = \dot{Y}\left(n - \frac{N}{2}\right) + e^{-j\frac{2\pi}{N}\left(n - \frac{N}{2}\right)} \dot{Z}\left(n - \frac{N}{2}\right). \quad (19)$$

Процесс вычисления 8-точечного ДПФ путем разбиения его на два 4-точечных ДПФ иллюстрируется на рис. 8.

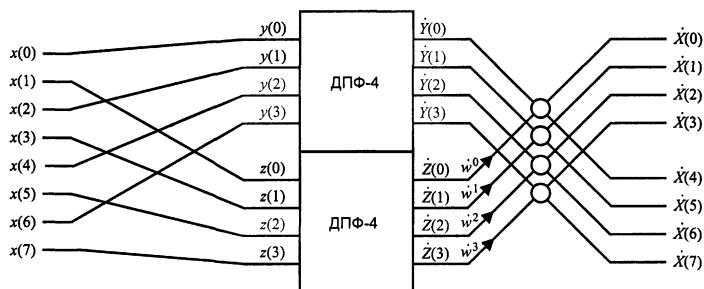


Рис. 8. Вычисление 8-точечного ДПФ с помощью двух 4-точечных ДПФ

Блоки, выполняющие на рис. 8 объединение результатов двух ДПФ, требуют дополнительных комментариев. Каждый такой блок имеет два входных и два выходных сигнала. Один из входных сигналов умножается на комплексную экспоненту w^k , после чего суммируется со вторым входным сигналом и вычитается из него, формируя таким образом два выходных сигнала. Это соответствует реализации формул (18) и (19). Данная операция получила название «бабочки» (butterfly). Расшифровка ее структуры представлена на рис. 9.

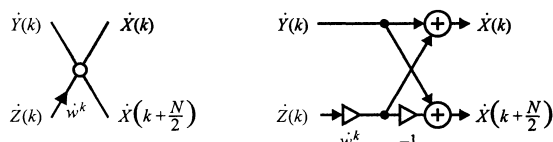


Рис. 9. Условное обозначение «бабочки» БПФ

с прореживанием по времени (слева) и ее структурная схема (справа)

Оценим количество операций, необходимое для вычисления ДПФ указанным способом. Каждое из двух ДПФ половинной размерности требует $N^2/4$ операций. Кроме того, при вычислении окончательных результатов каждый спектральный коэффициент $\dot{Z}(n)$ умножается на экспоненциальный комплексный множитель. Это добавляет еще $N/2$ операций. Итого получается $2N^2/4 + N/2 = N(N+1)/2$, что почти вдвое меньше, чем при вычислении ДПФ прямым способом.

Если $N/2$ тоже является четным числом (то есть если N делится на 4), можно продолжить описанную процедуру, выразив результат через четыре ДПФ размерности $N/4$. Это позволяет еще больше сократить число требуемых вычислительных операций.

Наибольшая степень ускорения вычислений может быть достигнута при $N = 2^k$ в этом случае деление последовательностей на две части можно продолжать до тех пор, пока не получатся двухэлементные последовательности, ДПФ которых рассчитывается вообще без использования операций умножения (достаточно вычислить сумму и разность двух отсчетов). Число требуемых при этом пар операций «умножение - сложение» можно оценить как $N \log_2(N)$. Таким образом, вычислительные затраты по сравнению с непосредственным использованием формулы (18) уменьшаются в $N/\log_2(N)$ раз. При

больших N это отношение становится весьма велико (например, $1024/\log_2(1024)=102,4$, то есть при $N = 1024$ достигается более чем 100-кратное ускорение).

VII. ПРАКТИЧЕСКАЯ ЧАСТЬ

Пример вычисления 8-точечного ДПФ с помощью алгоритма БПФ с прореживанием по времени

Алгоритм БПФ для 8-точечного ДПФ представляет собой трёхэтапную процедуру вычисления ДПФ по общей формуле (19) при $i=1, 2, 3$. Формирование начальных условий перед выполнением первого этапа заключается в трёхкратном разбиении исходной 8-точечной последовательности на группы чётных и нечётных отсчётов, а именно:

- 1) 8-точечная последовательность разбивается на две группы 4-точечных с чётными и нечётными отсчётами (см. рис. 8):

$[x(0), x(2), x(4), x(6)]$ - чётные отсчёты,

$[x(1), x(3), x(5), x(7)]$ - нечётные отсчёты;

- 2) каждая из двух 4-точечных последовательностей разбивается на две группы 2-точечных с чётными и нечётными отсчётами по порядку следования, считая от нуля, а именно:

последовательность $[x(0), x(2), x(4), x(6)]$:

$[x(0), x(4)]$ - чётные отсчёты,

$[x(2), x(6)]$ - нечётные отсчёты;

последовательность $[x(1), x(3), x(5), x(7)]$:

$[x(1), x(5)]$ - чётные отсчёты,

$[x(3), x(7)]$ - нечётные отсчёты;

- 3) каждая из четырёх 2-точечных последовательностей разбивается на два отсчёта - чётный и нечётный отсчётами по порядку следования, считая от нуля, т.е. на нулевой (чётный) и первый (нечётный), а именно:

последовательность $[x(0), x(4)]$:

$[x(0)]$ - чётные отсчёты,

$[x(4)]$ - нечётные отсчёты;

последовательность $[x(2), x(6)]$:

$[x(2)]$ - чётные отсчёты,

$[x(6)]$ - нечётные отсчёты;

последовательность $[x(1), x(5)]$:

$[x(1)]$ - чётные отсчёты,

$[x(5)]$ - нечётные отсчёты;

последовательность $[x(3), x(7)]$:

$[x(3)]$ - чётные отсчёты,

$[x(7)]$ - нечётные отсчёты.

После этого каждой паре отсчётов ДПФ $X_{2m}^0(0)$ и $X_{2m+1}^0(0)$, $m=0, 1, 2, 3$ последовательно присваиваются значения пары отсчётов прореженной 8-точечной последовательности, а именно (см. рис. 10)

$$\begin{cases} X_0^0(0) = x(0), \\ X_1^0(0) = x(4); \end{cases} \quad \begin{cases} X_2^0(0) = x(2), \\ X_3^0(0) = x(6); \end{cases}$$

$$\begin{cases} X_4^0(0) = x(1), \\ X_5^0(0) = x(5); \end{cases} \quad \begin{cases} X_6^0(0) = x(3), \\ X_7^0(0) = x(7). \end{cases}$$

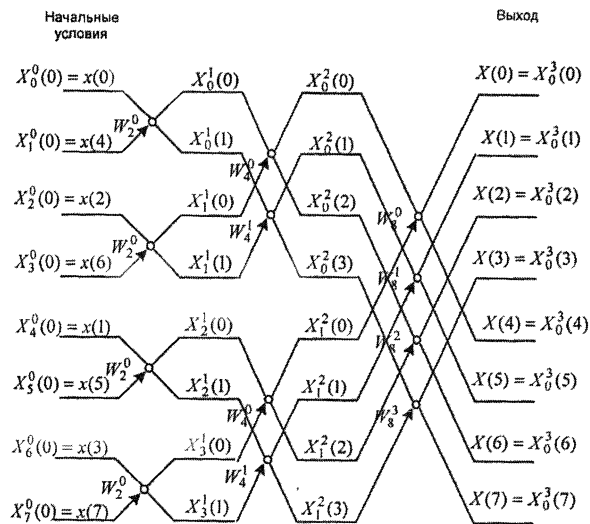


Рис. 10. Направленный граф алгоритма БПФ с прореживанием по времени для 8-точечного ДПФ

Далее выполняется трёхэтапный алгоритм БПФ, направленный граф которого приводится на рис. 10:

1) $i=1$

На первом этапе выполняется четыре операции «бабочка» при $m=0, 1, 2, 3$ с учётом начальных условий

$$\begin{cases} X_0^1(0) = x(0) + W_2^0 x(4), & X_1^1(0) = x(2) + W_2^0 x(6), \\ X_0^1(1) = x(0) - W_2^0 x(4); & X_1^1(1) = x(2) - W_2^0 x(6); \\ X_2^1(0) = x(1) + W_2^0 x(5), & X_2^1(0) = x(3) + W_2^0 x(7), \\ X_2^1(1) = x(1) - W_2^0 x(5); & X_2^1(1) = x(3) - W_2^0 x(7); \end{cases}$$

Здесь поворачивающий множитель оставлен исключительно для единообразия «бабочек»; в действительности $W_2^0 = 1$.

2) $i=2$

На втором этапе выполняется четыре операции «бабочка» при $m=0, 1$

$$\begin{cases} X_0^2(k) = X_0^1(k) + W_4^k X_1^1(k); \\ X_0^2(k+2) = X_0^1(k) - W_4^k X_1^1(k); \\ k = 0, 1, \\ X_1^2(k) = X_2^1(k) + W_4^k X_3^1(k); \\ X_1^2(k+2) = X_2^1(k) - W_4^k X_3^1(k); \\ k = 0, 1. \end{cases}$$

3) $i=3$

На третьем этапе выполняются четыре операции «бабочка» при $m=0$

$$\begin{cases} X_0^3(k) = X_0^2(k) + W_8^k X_1^2(k); \\ X_0^3(k+2) = X_0^2(k) - W_8^k X_1^2(k); \\ k = 0, 1, 2, 3. \end{cases}$$

На выходе имеем 8-точечное ДПФ, отсчёты которого следуют в естественном порядке

$$\begin{cases} X(k) = X_0^3(k); \\ k = 0, 1, \dots, N-1. \end{cases}$$

VIII. ЗАДАНИЕ НА РАБОТУ

Изучить основные положения теории вычисления ДПФ с помощью алгоритма БПФ с прореживанием по времени.

Изучить функции MATLAB дискретного преобразования Фурье.

Провести вычисления 8-точечного ДПФ с помощью алгоритма БПФ с прореживанием по времени.

IX. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Изучить основные положения теории вычисления ДПФ с помощью алгоритма БПФ с прореживанием по времени.
2. Изучить основные функции MATLAB, необходимые для расчёта ДПФ.
3. В соответствии с планом предложенным в параграфе III провести вычисления 8-точечного ДПФ с помощью алгоритма БПФ с прореживанием по времени с заданными преподавателем параметрами.
4. Проверить в Matlab результаты вычисления ДПФ.

VI. ОФОРМЛЕНИЕ ОТЧЕТА

Отчёт должен содержать:

- краткие теоретические сведения по выполняемой работе;
- алгоритмы и листинги программ расчёта ДПФ;
- выводы по проведённой работе.

VII. КОНТРОЛЬНЫЕ ВОПРОСЫ

VIII. БИБЛИОГРАФИЧЕСКИЙ СПИСОК

6. Конспект лекций по курсу.
7. Дьяконов В. MATLAB: учебный курс. – СПб: Питер, 2001. – 560 с.: ил.
8. Цифровая обработка сигналов / А.Б. Сергиенко – СПб.: Питер, 2003. – 604 с.: ил.

Синтез характеристик аналоговых фильтров и цифровых фильтров по аналоговому прототипу.

I. ЦЕЛЬ И ЗАДАЧИ РАБОТЫ

Изучение аналоговых фильтров. Расчёт аналоговых фильтров в Matlab. Применение аналоговых фильтров.

II. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Аналоговые фильтры

Теория линейных стационарных непрерывных систем широко используется при построении различных частотно-избирательных цепей. Аналоговым фильтрами называется частотно-избирательная цепь, обеспечивающая пропускание сигналов в определенных полосах частот и подавление в других. Область частот, в которой фильтр пропускает сигнал, называют полосой пропускания, а область частот, в которой ослабление входного сигнала велико - полосой задерживания.

Фильтры применяют для выделения требуемого сигнала из смеси полезных и нежелательных сигналов. В зависимости от взаимного расположения полосы пропускания и полосы задерживания различают следующие виды фильтров:

- *фильтры нижних частот* (ФНЧ; английский термин — low-pass filter), пропускающие частоты, меньшие некоторой частоты среза ω_0 ;
- *фильтры верхних частот* (ФВЧ; английский термин — high-pass filter), пропускающие частоты, большие некоторой частоты среза ω_0 ;
- *полосовые фильтры* (ПФ; английский термин — band-pass filter), пропускающие частоты в некотором диапазоне $\omega_1 \dots \omega_2$ (они могут также характеризоваться средней частотой $\omega_0 = (\omega_1 + \omega_2)/2$ и шириной полосы пропускания $\Delta\omega = \omega_2 - \omega_1$);
- *режекторные фильтры* (другие возможные названия — заграждающий фильтр, фильтр-пробка, полосно-задерживающий фильтр; английский термин — band-stop filter), пропускающие на выход все частоты, кроме лежащих в некотором диапазоне $\omega_1 \dots \omega_2$ (они тоже могут характеризоваться средней частотой $\omega_0 = (\omega_1 + \omega_2)/2$ и шириной полосы задерживания $\Delta\omega = \omega_2 - \omega_1$).

Идеальная форма АЧХ фильтров этих четырех типов показана на рис. 1. Однако такая идеальная форма АЧХ не может быть физически реализована. Поэтому в теории аналоговых фильтров разработан ряд методов аппроксимации прямоугольных АЧХ.

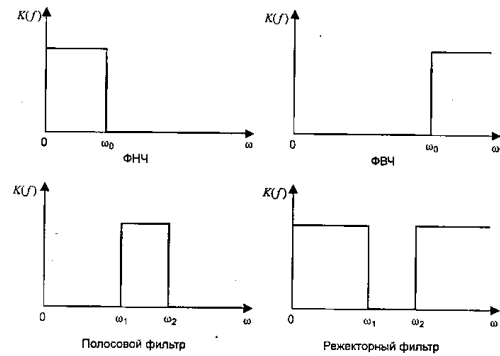


Рис. 1. АЧХ фильтров различного типа.

Фильтр Баттерворта

Функция передачи фильтра Баттерворта (Butterworth filter) не имеет нулей, а ее полюсы равномерно расположены на s -плоскости в левой половине окружности единичного радиуса:

$$H(p) = \frac{1}{\prod_{k=1}^n (p - p_k)}$$

где полюса s_k определяются следующим образом:

$$p_k = \cos\left(\pi \frac{n-1+2k}{2n}\right) + j \sin\left(\pi \frac{n-1+2k}{2n}\right).$$

Благодаря такому размещению полюсов формула для АЧХ фильтра Баттерворта оказывается очень простой:

$$K(\omega) = \frac{1}{\sqrt{1 + \left(\frac{\omega}{\omega_0}\right)^{2n}}},$$

где частота среза ω_0 (для фильтра-прототипа она равна 1 рад/с), n - порядок фильтра.

Коэффициент передачи на нулевой частоте равен 1, а на частоте среза независимо от порядка фильтра составляет $1/\sqrt{2} \approx 0.707 \approx 3\text{дБ}$. При АЧХ $\omega \rightarrow \infty$ стремится к нулю.

АЧХ фильтра Баттерворта является максимально плоской при $\omega=0$ и $\omega \rightarrow \infty$. Это означает, что в данных точках равны нулю $2n - 1$ производных АЧХ по частоте.

В целом АЧХ монотонно спадает от единицы до нуля при изменении частоты от нуля до бесконечности.

Фильтр Чебышева первого рода

Функция передачи фильтра Чебышева первого рода (Chebyshev type I filter) также не имеет нулей, а ее полюсы расположены в левой половине эллипса на s -плоскости:

$$H(p) = \frac{1}{\prod_{k=1}^n (p - p_k)}$$

где полюса s_k определяются следующим образом:

$$p_k = j\omega_1 \{ \cos(\alpha_k) \cosh(\beta_k) - j \sin(\alpha_k) \sinh(\beta_k) \},$$

$$\alpha_k = \frac{(2k+1)\pi}{2n}; \beta_k = \frac{1}{n} \operatorname{arsh}\left(\frac{1}{\varepsilon}\right).$$

АЧХ фильтра Чебышева первого рода описывается следующим образом:

$$K(\omega) = \frac{1}{\sqrt{1 + \varepsilon^2 T_n^2\left(\frac{\omega}{\omega_0}\right)}}$$

Здесь ω - частота среза, $T_n(x)$ - полином Чебышева n -го порядка, n - порядок фильтра, ε - параметр, определяющий величину пульсаций АЧХ в полосе пропускания.

Полином Чебышева $T_n(x)$ при $|x| \leq 1$ колеблется в диапазоне $-1 \dots +1$, а при $|x| > 1$ неограниченно возрастает по абсолютной величине. Поэтому АЧХ фильтра Чебышева первого рода в полосе пропускания (при $|\omega| \leq \omega_0$) колеблется между значениями $1/\sqrt{1+\varepsilon^2}$ и 1, а вне полосы пропускания (при $|\omega| > \omega_0$) монотонно затухает до нуля.

Коэффициент передачи на нулевой частоте равен 1 при нечетном порядке фильтра и $1/\sqrt{1+\varepsilon^2}$ при четном. На частоте среза коэффициент передачи фильтра равен $1/\sqrt{1+\varepsilon^2}$, то есть уровню пульсаций АЧХ в полосе пропускания. При $\omega \rightarrow \infty$ АЧХ стремится к нулю.

По сравнению с фильтром Баттерворта того же порядка фильтр Чебышева обеспечивает более крутой спад АЧХ в области перехода от полосы пропускания к полосе задерживания.

Значение параметра ε и уровень пульсаций R_p (в децибелах) связаны следующим образом:

$$R_p = 20 \lg\left(\sqrt{1 + \varepsilon^2}\right) = 10 \lg(1 + \varepsilon^2) \text{ дБ},$$

$$\varepsilon = \sqrt{10^{R_p/10} - 1}.$$

При $\omega \rightarrow \infty$ АЧХ фильтра Чебышева первого рода является максимально плоской.

Фильтр Чебышева второго рода

Функция передачи фильтра Чебышева второго рода (Chebyshev type II filter), в отличие от предыдущих случаев, имеет и нули, и полюсы. Она связана с функцией передачи фильтра Чебышева первого рода следующим образом:

$$H_2(p) = 1 - H_1(1/p).$$

Здесь $H_1(p)$ и $H_2(p)$ функции передачи фильтров Чебышева первого и второго рода соответственно.

Полюсы функции передачи фильтров Чебышева первого и второго рода (p_{1i} и p_{2i} соответственно) связаны друг с другом соотношением $p_{2i} = 1/p_{1i}$.

По этой причине фильтры Чебышева второго рода называют еще *инверсными фильтрами Чебышева* (inverse Chebyshev filter).

АЧХ фильтра Чебышева второго рода описывается следующим образом:

$$K(\omega) = \frac{1}{\sqrt{1 + \frac{\varepsilon^2}{T_n^2(\omega/\omega_0)}}}$$

Здесь ω - частота среза, $T_n(x)$ - полином Чебышева n -го порядка, n - порядок фильтра, ε - параметр, определяющий величину пульсаций АЧХ в полосе задерживания.

В результате указанного выше преобразования функции передачи АЧХ фильтра Чебышева второго рода ведет себя следующим образом: в полосе пропускания она монотонно затухает, а в полосе задерживания колеблется между нулем и значением $1/\sqrt{1+\varepsilon^2}$.

Коэффициент передачи фильтра на нулевой частоте равен 1, на частоте среза — заданному уровню пульсаций в полосе задерживания. При $\omega \rightarrow \infty$ коэффициент передачи равен нулю при нечетном порядке фильтра и уровню пульсаций - при четном.

Значение параметра ε и уровень пульсации R_s (в децибелах) связаны следующим образом:

$$R_s = 20 \lg(\sqrt{1+\varepsilon^2}) = 10 \lg(1+\varepsilon^2) \text{ дБ},$$

$$\varepsilon = \sqrt{10^{R_s/10} - 1}.$$

При $\omega=0$ АЧХ фильтра Чебышева второго рода является максимально плоской.

Фильтр Бесселя

В отличие от фильтров предыдущих типов, фильтры Бесселя (Bessel filter) не аппроксимируют прямоугольную АЧХ - их АЧХ по форме близка к гауссовой кривой (точнее, стремится к ней с ростом порядка фильтра). Практическая ценность фильтров Бесселя определяется тем, что для них зависимость группового времени задержки от частоты является максимально гладкой в точке $\omega=0$ и групповая задержка очень мало меняется в полосе пропускания.

Функция передачи фильтра Бесселя имеет только полюсы, лежащие на окружности с центром в положительной области вещественной оси. Сама функция передачи имеет следующий вид:

$$H(p) = \frac{d_0}{\sum_{k=0}^n d_k p^k}.$$

Коэффициенты полинома знаменателя рассчитываются по следующей формуле:

$$d_k = \frac{(2n-k)!}{2^{n-k} k! (n-k)!}.$$

Расчёт аналоговых фильтров в MATLAB.

Для расчета аналогового фильтра необходимо выполнить две основные операции: рассчитать ФНЧ-прототип и преобразовать его к нужному типу фильтра с заданными частотами среза.

Расчёт ФНЧ-прототипа

В MATLAB расчёт аналоговых фильтров-прототипов для рассмотренных фильтро производится с помощью следующих функций:

- $[z,p,k]=\text{butter}(n)$ – для расчёта аналогового фильтра-прототипа Баттерворта;
- $[z,p,k]=\text{cheb1ap}(n,R_p)$ – для расчёта аналогового фильтра-прототипа Чебышева I-го рода;
- $[z,p,k]=\text{cheb2ap}(n,R_s)$ – для расчёта аналогового фильтра-прототипа Чебышева II-го рода;
- $[z,p,k]=\text{besselap}(n)$ – для расчёта аналогового фильтра-прототипа Бесселя.

Входные параметры для соответствующих фильтров-прототипов:

n – порядок фильтра (целочисленный);

R_p – уровень пульсаций в полосе пропускания (в дБ);

R_s – уровень пульсаций в полосе задерживания (в дБ).

Все функции MATLAB для расчета аналоговых прототипов возвращают векторы-столбцы нулей и полюсов функции передачи, а также значение коэффициента усиления.

Преобразование способов описания линейных цепей

Необходимость в преобразовании описаний часто возникает из-за того, что функции расчета цепей (такие как рассматриваемые функции расчета фильтров-прототипов) дают результат в одной форме, а функция, например, построения частотной характеристики требует задания входных параметров в другой форме.

Формы описания цепей в именах функций обозначаются следующим образом:

tf - коэффициенты полиномов числителя и знаменателя функции передачи (transfer function);

zp - нули и полюсы (zeros and poles);

ss - описание в пространстве состояний (state-space).

Далее кратко рассматриваются конкретные функции преобразования описаний цепей. Для входных и выходных параметров используются следующие обозначения:

- функция передачи:

b - вектор-строка коэффициентов (в порядке убывания степеней) числителя функции передачи;

a - вектор-строка коэффициентов (в порядке убывания степеней) знаменателя функции передачи;

- нули и полюсы:

z - вектор нулей (столбец);

p - вектор полюсов (столбец);

k - коэффициент усиления (скаляр);

- пространство состояний:

A - квадратная матрица связи вектора состояния и его производной;

B - вектор-столбец связи входного сигнала и производной вектора состояния;

C - вектор-строка связи выходного сигнала и вектора состояния;

D - скалярный коэффициент связи выходного и входного сигналов.

Функция *tf2zp* преобразует наборы коэффициентов полиномов числителя и знаменателя функции передачи в векторы нулей и полюсов, рассчитывая также значение общего коэффициента усиления:

$$[z, p, k] = \text{tf2zp}(b, a)$$

Функция *zp2tf* является обратной по отношению к функции *tf2zp*: она осуществляет преобразование коэффициента усиления, а также векторов нулей и полюсов функции передачи в коэффициенты полиномов ее числителя и знаменателя:

$$[b, a] = \text{zp2tf}(z, p, k)$$

Функция *tf2ss* преобразует наборы коэффициентов полиномов числителя и знаменателя функции передачи в параметры представления цепи в пространстве состояний:

$$[A, B, C, D] = \text{tf2ss}(b, a)$$

Функция *ss2tf* является обратной по отношению к функции *tf2ss*: она преобразует параметры пространства состояний в коэффициенты полиномов функции передачи цепи:

$$[b, a] = ss2tf(A, B, C, D);$$

Функция *zp2ss* преобразует нули, полюсы и коэффициент усиления цепи в ее параметры пространства состояний:

$$[A, B, C, D] = zp2ss(z, p, k)$$

Функция *ss2zp* является обратной по отношению к функции *zp2ss*, преобразуя параметры пространства состояний в нули, полюсы и коэффициент усиления цепи:

$$[z, p, k] = ss2zp(A, B, C, D)$$

Функция *residue*

Идентификатор последней функции, позволяющей преобразовывать описания линейных цепей, выпадает из общего ряда. Это связано с тем, что данное преобразование не является специфическим для обработки сигналов - оно сводится к разложению дробно-рациональной функции на простейшие дроби и часто применяется в математике. По этой же причине данная функция относится не к пакету Signal Processing, а к базовой библиотеке MATLAB.

Преобразование функции передачи, заданной в виде коэффициентов полиномов числителя и знаменателя, в сумму простых дробей производится с помощью функции *residue*. Она же осуществляет и обратное преобразование; нужное направление преобразования выбирается в зависимости от числа входных параметров.

При двух входных параметрах производится разложение функции передачи на простые дроби:

$$[r, p, k] = residue(b, a)$$

Здесь *b* и *a* — коэффициенты полиномов числителя и знаменателя функции передачи соответственно. Выходные параметры — векторы-столбцы полюсов (*p*) и соответствующих им вычетов (*r*), а также строка коэффициентов целой части *k*.

Преобразования фильтров-прототипов

Следующий этап после расчета фильтра-прототипа - его преобразование с целью получения фильтра заданного вида с требуемыми частотами среза. Для этого используются приведенные ниже четыре функции MATLAB. Принцип составления их имен следующий: сначала идет сокращение *lp*, означающее, что исходным фильтром является ФНЧ (low-pass), потом следует символ преобразования 2 и в конце стоит обозначение типа результирующего фильтра:

- lp2lp - изменение частоты среза ФНЧ (low-pass);
- lp2hp - преобразование ФНЧ в ФВЧ (high-pass);
- lp2bp - преобразование ФНЧ в полосовой фильтр (band-pass);
- lp2bs - преобразование ФНЧ в режекторный фильтр (band-stop).

Все эти функции могут преобразовывать фильтры, заданные двумя способами - в виде коэффициентов полиномов числителя и знаменателя функции передачи либо в пространстве состояний. Различаются эти два варианта по числу входных и выходных параметров.

Изменение частоты среза ФНЧ

Изменение частоты среза ФНЧ-прототипа сводится к простому масштабированию частотной оси и выполняется путем следующей замены переменной s в выражении для функции передачи: $s \rightarrow \frac{s}{\omega_0}$, где ω_0 - требуемая частота среза ФНЧ. Такое преобразование производится функцией lp2lp:

$$[bl, al] = lp2lp(b, a, w0)$$

$$[A1, B1, C1, D1] = lp2lp(A, B, C, D, w0)$$

Входными параметрами функции являются описание фильтра (в виде коэффициентов полиномов числителя и знаменателя функции передачи - b, a , или в пространстве состояний - A, B, C, D) и требуемая частота среза $w0$.

Возвращаемый результат - пересчитанные параметры фильтра.

Преобразование ФНЧ в ФВЧ

Преобразование ФНЧ-прототипа в ФВЧ требует инверсии частотной оси и выполняется путем следующей замены переменной s в выражении для функции передачи:

$$s \rightarrow \frac{s}{\omega_0}, \text{ где } \omega_0 - \text{требуемая частота среза ФВЧ.}$$

Такое преобразование производится функцией lp2hp:

$$[bl, al] = lp2hp(b, a, w0)$$

$$[A1, B1, C1, D1] = lp2hp(A, B, C, D, w0)$$

Входными параметрами функции являются описание фильтра (в виде коэффициентов полиномов числителя и знаменателя функции передачи - b, a или в пространстве состояний - A, B, C, D) и требуемая частота среза $w0$.

Возвращаемый результат - пересчитанные параметры фильтра.

Преобразование ФНЧ в полосовой фильтр

Преобразование ФНЧ-прототипа в полосовой фильтр требует более сложной трансформации частотной оси, чем в предыдущих случаях. Так, нулевая и бесконечная частоты должны преобразовываться в бесконечное значение на частотной оси ФНЧ-прототипа (там, где его коэффициент передачи стремится к нулю). Частоты, соответствующие краям требуемой полосы пропускания, должны после преобразования давать значения ± 1 , равные частоте среза ФНЧ-прототипа. Наконец, преобразование должно выполняться с помощью дробно-рациональной функции, чтобы сохранить дробно-рациональную структуру функции передачи.

Перечисленным требованиям удовлетворяет следующая замена переменной s :

$$s \rightarrow Q \frac{\left(\frac{s}{\omega_0}\right)^2 + 1}{\frac{s}{\omega_0}},$$

где $\omega_0 = \sqrt{\omega_1 \omega_2}$, $Q = \omega_0 / (\omega_2 - \omega_1)$, ω_1 и ω_2 соответственно нижняя и верхняя границы полосы пропускания фильтра.

Такое преобразование производится функцией `lp2bp`:

`[bl, al] = lp2bp(b, a, w0, Bw)`

`[A1, B1, C1, D1] = lp2bp(A, B, C, D, w0, Bw)`

Входными параметрами функции являются описание фильтра (в виде коэффициентов полиномов числителя и знаменателя функции передачи - b , a или в пространстве состояний - A , B , C , D), средняя частота $w0$ и ширина Bw полосы пропускания фильтра (в рад/сек).

Возвращаемый результат - пересчитанные параметры фильтра.

Преобразование ФНЧ в режсекторный фильтр

Для преобразования ФНЧ-прототипа в режсекторный фильтр трансформация частотной оси должна быть обратной по отношению к предыдущему случаю. Нулевая и бесконечная частоты должны преобразовываться в нулевое значение на частотной оси ФНЧ-прототипа (там, где коэффициент передачи велик). Частоты, соответствующие краям требуемой полосы задерживания, должны после преобразования давать значения ± 1 , равные частоте среза ФНЧ-прототипа. Кроме того, некоторое значение частоты в полосе задерживания должно преобразовываться в бесконечность (там, где коэффициент передачи ФНЧ-прототипа стремится к нулю). Наконец, преобразование должно выполняться с помощью дробно-рациональной функции, чтобы сохранить дробно-рациональную структуру функции передачи.

Перечисленным требованиям удовлетворяет следующая замена переменной s :

$$s \rightarrow \frac{s/\omega_0}{Q \left[\left(\frac{s/\omega_0}{\omega_2 - \omega_1} \right)^2 + 1 \right]},$$

где $\omega_0 = \sqrt{\omega_1 \omega_2}$, $Q = \omega_0 / (\omega_2 - \omega_1)$, ω_1 и ω_2 соответственно нижняя и верхняя границы полосы задерживания фильтра.

Такое преобразование выполняется функцией `lp2bs`:

$$[bl, al] = lp2bs(b, a, w0, Bw)$$

$$[Al, Bl, Cl, D1] = lp2bs(A, B, C, D, w0, Bw)$$

Входными параметрами функции являются описание фильтра (в виде коэффициентов полиномов числителя и знаменателя функции передачи - b , a или в пространстве состояний - A , B , C , D), средняя частота $w0$ и ширина Bw полосы задерживания фильтра (в радианах в секунду).

Возвращаемый результат - пересчитанные параметры фильтра.

Все сказанное выше применительно к полосовым фильтрам о связи средней частоты с частотами среза и об асимметрии характеристик справедливо и для режекторных фильтров.

Рассмотренная выше требуемая последовательность действий оформлена в виде следующих функций MATLAB:

`butter(n, w0, type, 's')` - расчет фильтров Баттерворта;

`cheby1(n, Rp, w0, type, 's')` - расчет фильтров Чебышева первого рода;

`cheby2(n, Rs, w0, type, 's')` - расчет фильтров Чебышева второго рода;

`besself(n, w0, type)` - расчет фильтров Бесселя.

Параметры всех функций задаются одинаково.

Перечисленные функции, за исключением функции `besself`, позволяют рассчитывать как аналоговые, так и дискретные фильтры. Признаком аналогового расчета служит строка 's', использованная в качестве последнего входного параметра. Параметры n , Rp , Rs (их состав зависит от типа фильтра) — это параметры фильтра-прототипа: n — порядок фильтра, Rp — уровень пульсаций в полосе пропускания (в децибелах), Rs — уровень пульсаций в полосе задерживания (в децибелах).

Параметры $w0$ и $type$ используются совместно для задания типа фильтра и значений его частот среза (в радианах в секунду):

- ФНЧ: w_0 - скаляр, параметр `type` отсутствует;
- ФВЧ: w_0 - скаляр, `type='high'`;
- полосовой фильтр: w_0 - двухэлементный вектор частот среза $[w_1 \ w_2]$, параметр `type` отсутствует;
- режекторный фильтр: w_0 - двухэлементный вектор частот среза $[w_1 \ w_2]$, `type='stop'`.

В зависимости от того, сколько выходных параметров указано при вызове, функции могут возвращать результаты расчета в виде коэффициентов полиномов числителя и знаменателя функции передачи (два выходных параметра), нулей и полюсов (три выходных параметра) либо параметров пространства состояний (четыре выходных параметра):

$[b, a] = \dots$

$[z, p, k] = \dots$

$[A, B, C, D] = \dots$

С учетом всего сказанного перечислим действия, выполняемые функциями расчета аналоговых фильтров:

1. Производится расчет фильтра-прототипа с заданными параметрами АЧХ.
2. Полученные нули и полюсы преобразуются в параметры пространства состояний.
3. Производится преобразование фильтра-прототипа к требуемому типу с заданными частотами среза.
4. Выполняется преобразование описания фильтра к заданному при вызове виду.

Выбор порядка фильтра

Рассмотренные выше функции расчета фильтров требуют задания в качестве входных параметров порядка фильтра и его частоты среза. При этом понятие частоты среза для фильтров разных типов определяется по-разному. Однако исходными данными при разработке фильтров, как правило, являются другие параметры: частотные границы полос пропускания (ω_p) и задерживания (ω_s), а также допустимая неравномерность АЧХ в полосе пропускания (R_p) и минимально необходимое затухание в полосе задерживания (R_s). Серые области на рисунке демонстрируют допуски, в которые должна укладываться АЧХ фильтра в полосах пропускания и задерживания.

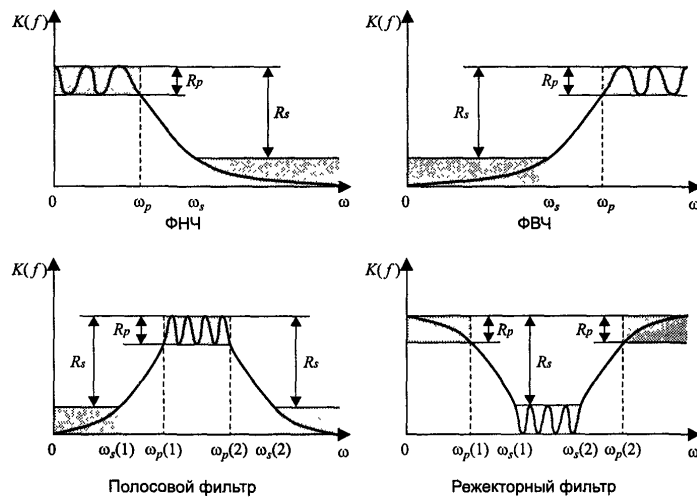


Рис. 2. Задание исходных параметров для расчета ФНЧ, ФВЧ, полосовых и режекторных фильтров

Выбрать минимально необходимый порядок фильтра позволяют следующие од-
нотипные функции пакета Signal Processing:

$[n, Wn] = \text{buttord}(Wp, Ws, Rp, Rs, 's')$

$[n, Wn] = \text{cheblord}(Wp, Ws, Rp, Rs, 's')$

$[n, Wn] = \text{cheb2ord}(Wp, Ws, Rp, Rs, 's')$

Входной параметр R_p - допустимый уровень пульсаций в полосе пропускания (в децибелах), R_s - минимально необходимое затухание в полосе задерживания (в децибелах). Параметры W_p и W_s задают границы полос пропускания и задерживания, способ задания этих параметров зависит от типа проектируемого фильтра:

- ФНЧ: W_p и W_s — числа, при этом должно выполняться неравенство $W_p < W_s$;
- ФВЧ: W_p и W_s — числа, при этом должно выполняться неравенство $W_p > W_s$;
- полосовой фильтр: W_p и W_s — двухэлементные векторы, при этом должны выполняться неравенства $W_s(1) < W_p(1) < W_p(2) < W_s(2)$;
- режекторный фильтр: W_p и W_s — двухэлементные векторы, при этом должны выполняться неравенства $W_p(1) < W_s(1) < W_s(2) < W_p(2)$.

Значения параметров W_p и W_s были обозначены на рис. 2 как ω_p и ω_s соответственно.

Выходными параметрами являются минимально необходимый для выполнения заданных требований порядок фильтра n и частота среза фильтра Wn . Эти параметры должны затем использоваться при вызове функции расчета фильтра. Возврат значения Wn избавляет пользователя от забот, связанных с тем, что при расчете разных фильтров понятие частоты среза имеет разный смысл.

Поскольку порядок фильтра - величина целочисленная, то обычно оказывается, что фильтр минимально необходимого порядка обеспечивает некоторый запас по исходным параметрам. Этот запас можно использовать по-разному - либо сделать пульсации в полосе пропускания точно равными заданным, но увеличить затухание в полосе задерживания, либо точно выдержать заданное затухание в полосе задерживания, уменьшив при этом пульсации в полосе пропускания. Поведение функций выбора порядка фильтра в этом аспекте определяется тем, что при расчете фильтра должны будут использоваться те же параметры пульсаций R_p и R_s , что и при выборе порядка фильтра. Поэтому для фильтров Баттерворта и Чебышева первого рода будет увеличиваться затухание в полосе задерживания, для фильтров Чебышева второго рода - уменьшаться пульсации в полосе пропускания.

III. ПРАКТИЧЕСКАЯ ЧАСТЬ

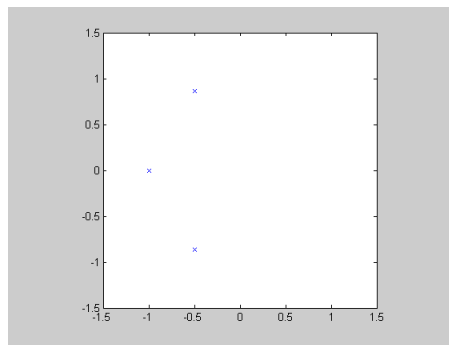
Рассмотрим реализацию аналогового фильтра верхних частот в MATLAB на примере фильтра Баттерворта 3-го порядка с частотой среза $\omega_0=10$ рад/сек.

1. Для расчёта характеристик фильтра-прототипа создаём m-файл. Листинг программы представлен на рис. 3.

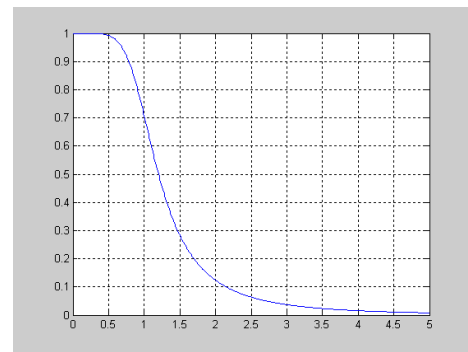
```
n=3
[z,p,k]=buttap(n)           % порядок фильтра
plot(p,'x')                % нули и полюса прототипа
axis equal
axis([-1.5 1.5 -1.5 1.5])
w=0:0.01:5;
[b,a]=zp2tf(z,p,k)         % коэффициенты функции
                             % передачи
h=freqs(b,a,w);            % комплексный коэффициент
                             % передачи
figure                     % вызов окна построения
                             % графика
plot(w,abs(h)), grid       % график АЧХ
                             % график ФЧХ
```

Рис. 3. Листинг программы расчёта фильтра-прототипа

Результаты расчёта представлены на рис. 4.

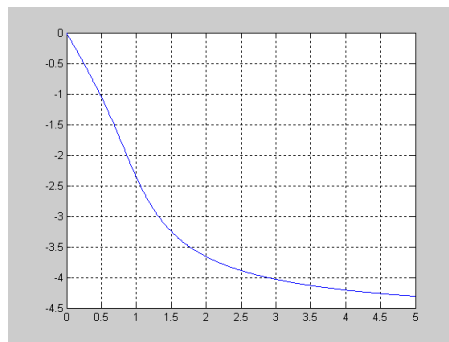


а



б

Рис. 4. Характеристики фильтра-прототипа Баттерворта 3-го порядка: а – расположение полюсов на комплексной плоскости, б – АЧХ, в – ФЧХ



в

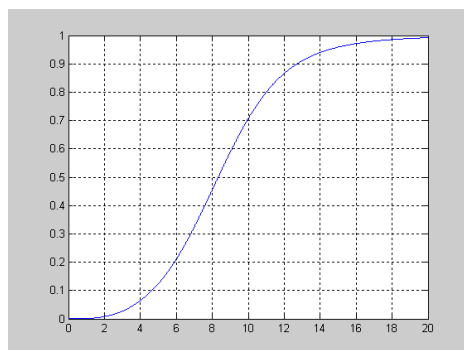
2. Преобразуем ФНЧ в ФВЧ с заданной частотой среза. Листинг программы представлен на рис. 5.

```
w0=10
[b1, a1]=lp2hp(b, a, w0)
h=freqs(b1,a1,w);
figure
plot(w,abs(h)), grid
figure
plot(w,unwrap(angle(h))), grid
```

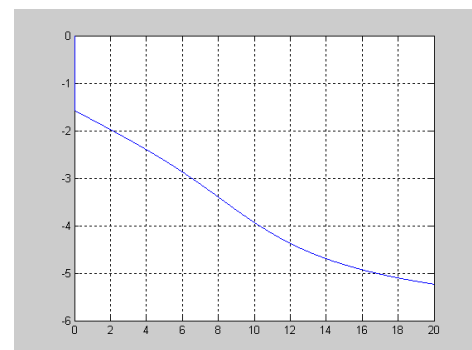
```
% заданная частота среза
% фильтра
% преобразование ФНЧ-
% прототипа к ФВЧ
% комплексный коэффициент
% передачи
% вызов окна построения
% графика
% график АЧХ
```

```
% график ФЧХ
```

Рис. 5. Листинг программы преобразования ФНЧ-прототипа в ФВЧ. Результаты представлены на рис. 6.



а



б

Рис. 6. Характеристики ФВЧ Баттерворта 3-го порядка: а – АЧХ, б – ФЧХ

3. Результаты расчёта фильтра проверяем на модели созданной в SIMULINK. На рис.7. представлена модель системы с рассчитанным фильтром.

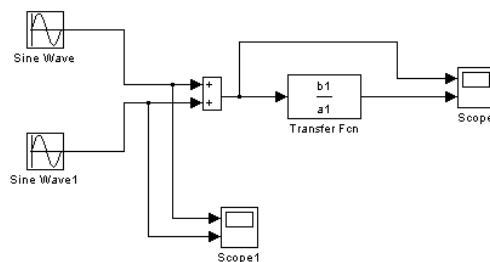


Рис. 7 Модель системы рассчитанным фильтром.

IV. ЗАДАНИЕ НА РАБОТУ

Изучить аналоговые фильтры.

Изучить основные функции MATLAB, необходимые для расчёта аналоговых фильтров.

Рассчитать аналоговый фильтр.

V. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Изучить аналоговые фильтры.
2. Изучить основные функции MATLAB, необходимые для расчёта аналоговых фильтров.
3. В соответствии с планом предложенным в параграфе IV рассчитать аналоговый фильтр с заданными преподавателем параметрами.
4. Сравнить результаты проведённого расчёта фильтра с результатами расчёта при использовании специальных функций MATLAB непосредственного расчёта фильтров.
5. Произвести выбор порядка фильтра по заданным преподавателем параметрам.

VI. ОФОРМЛЕНИЕ ОТЧЕТА

Отчёт должен содержать:

- краткие теоретические сведения по выполняемой работе;
- алгоритмы и листинги программ расчёта фильтра;
- выводы по проведённой работе.

VII. КОНТРОЛЬНЫЕ ВОПРОСЫ

7. Дайте определения фильтрам НЧ, ВЧ,...
8. Основные характеристики фильтров Баттерворта, Чебышева, Бесселя...
9. Какой из перечисленных типов фильтров имеет наиболее крутой спад АЧХ в области заграждения.
10. Какой из перечисленных типов фильтра имеет минимальный фазовый сдвиг в рабочей полосе частот.
11. Перечислите параметры, которые необходимо задать при расчете фильтров НЧ, ВЧ, режекторных ...

VIII. БИБЛИОГРАФИЧЕСКИЙ СПИСОК

9. Конспект лекций по курсу.
10. Дьяконов В. MATLAB: учебный курс. – СПб: Питер, 2001. – 560 с.: ил.
11. Цифровая обработка сигналов / А.Б. Сергиенко – СПб.: Питер, 2003. – 604 с.: ил.

Синтез цифровых корректирующих устройств

Цель работы:

1. Синтез Z-передаточной функции корректирующего устройства с помощью пакета Matlab.
2. Программирование Z-передаточной функции.
3. Получение основных характеристик скорректированного ЦЭСП.

1. Техническое описание

Классический способ синтеза является синтез методом частотных характеристик. Однако в системах с цифровым вычислителем непосредственно использовать этот простой метод не удастся. Один из путей решения этой проблемы – использование Z-преобразования, далее выполнение билинейного преобразования и переход к псевдочастотным характеристикам, после чего получение передаточной функции фильтра и обратный переход к Z-передаточным функциям. Безусловно, этот путь является правильным, но, в то же время, достаточно трудоемким. Однако, если принять некоторые допущения, то можно значительно упростить эту процедуру.

На рисунке 1. представлена структурная схема привода с использованием аналогового фильтра. Передаточную функцию фильтра получим, используя исходную и желаемую ЛАЧХ разомкнутой системы.

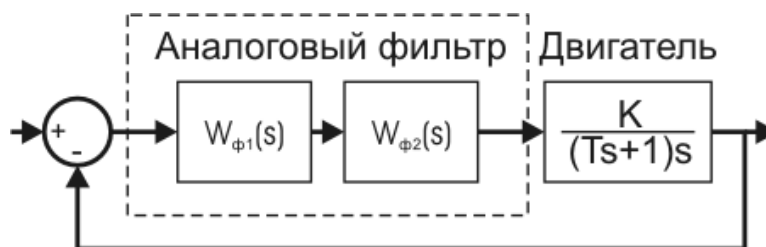


Рисунок 1. Упрощенная модель привода с аналоговым фильтром.

Исходная ЛАЧХ – это ЛАЧХ двигателя, который представлен апериодическим звеном и интегратором. На рисунке 2 представлена ЛАЧХ исходной (сплошная) и желаемой (штриховая) системы, а также показаны частоты всех изломов ($\omega_1, \omega_1 \dots$). Передаточная функция фильтра получается путем вычитания исходной ЛАЧХ из желаемой.

После того как получена передаточная функция фильтра, производим Z-преобразование и получим Z-функцию фильтра. Однако надо помнить об одном ограничении: такт квантования в системе должен быть хотя бы в 2 раза меньше самой маленькой постоянной времени системы. На рисунке 3 представлена эквивалентная схема системы, но уже с цифровым фильтром.

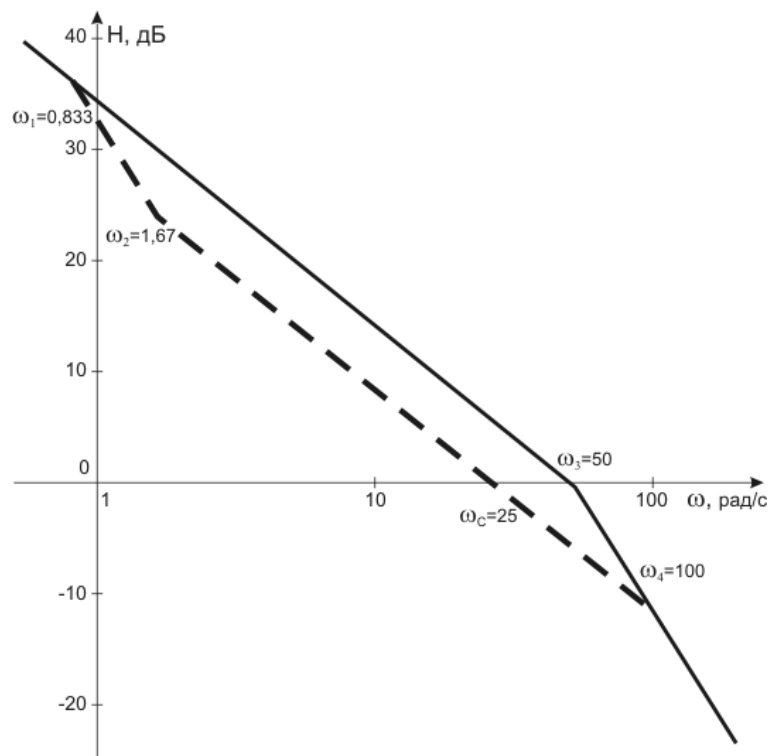


Рисунок 2. ЛАЧХ исходной (сплошная) и желаемой (штриховая) системы.

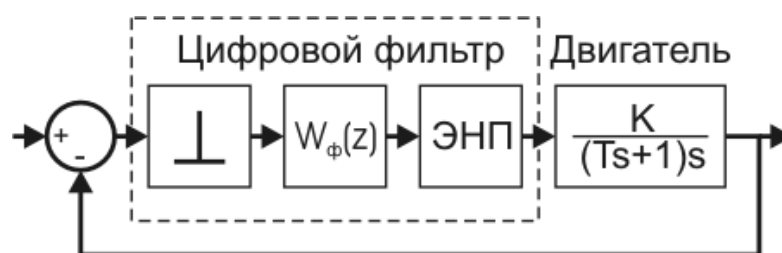


Рисунок 3. Упрощенная модель привода с цифровым фильтром.

Далее необходимо запрограммировать Z-функцию фильтра. Существует несколько способов выбора переменных состояния; будем использовать прямое, параллельное или последовательное программирование.

В результате получаем структурную схему, вводим переменные состояния и записываем в виде системы разностных уравнений. Далее необходимо выписать главную

матрицу A, матрицу-столбец B, матрицу-строку C и матрицу D. Полученные матрицы вводятся в программу Scope.exe (смотри рисунок 4).

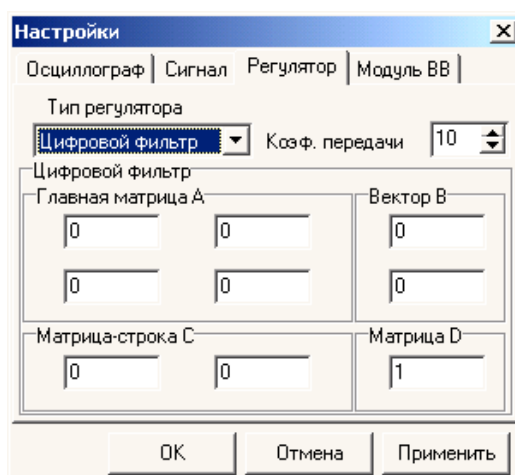


Рисунок 4. Окно ввода коэффициента передачи и матриц фильтра.

2. Приборы и оборудование

- Персональный компьютер с установленной системой Matlab;
- Приложение Scope.exe;
- Цифровой следящий привод.

3. Порядок выполнения работы.

- Передаточная функция двигателя имеет вид:

$$W_{\text{дв}}(s) = \frac{7.14}{(19.2 \cdot 10^{-3} s + 1)s}$$

По рисунку 2 необходимо определить величину коэффициента передачи по контуру. А затем получить коэффициент передачи фильтра K.

- После этого открываем модель Simulink и вводим полученный коэффициент передачи (смотри рисунок 5). Сохраняем переходный процесс в отчет.

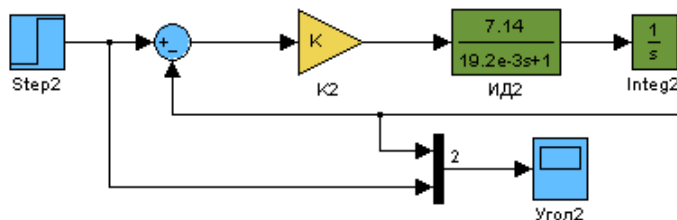


Рисунок 5. Упрощенная модель с П-регулятором.

- Необходимо получить передаточную функцию фильтра вычитанием исходной ЛАЧХ из желаемой (смотри рисунок 2). В результате должна получиться передаточная функция вида:

$$W(s) = \frac{(T2s + 1)(T3s + 1)}{(T1s + 1)(T4s + 1)} \quad (1)$$

- Ввести полученный аналоговый фильтр в модель Simulink (смотри рисунок 6).
Сохранить переходный процесс в отчет.

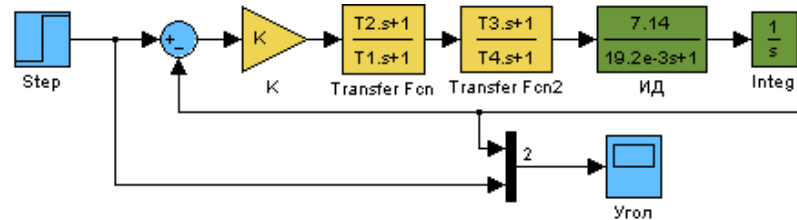


Рисунок 6. Модель с аналоговым фильтром.

- В командной строке Matlab создать переменную s – оператор дифференцирования. Для этого необходимо воспользоваться функцией `tf` (transfer function - передаточная функция): `s = tf('s')`
- Далее вводим передаточную функцию в виде: $W_s = \frac{(T2*s+1)*(T1*s+1)}{(T3*s+1)*(T4*s+1)}$
- где $T1, T2 \dots T4$ – конкретные числа из формулы (1).
- Далее производим Z-преобразование с помощью функции `c2d` (continues to discrete): $W_z = c2d(W_s, T)$
где W_s – введенная ранее переменная; T – такт квантования, с. Для совместимости со стендом примем $T = 5$ мс.
- Вводим полученный цифровой фильтр в модель Simulink (смотри рисунок 7).
Сохраняем переходный процесс в отчет.

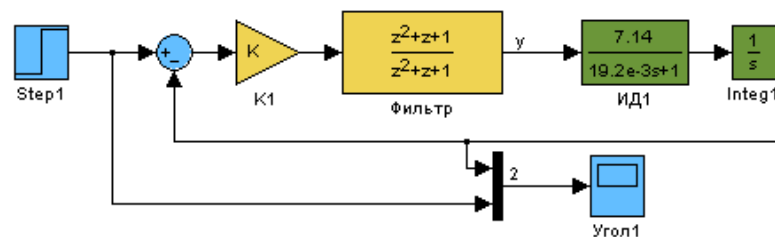


Рисунок 7. Модель с цифровым фильтром.

- После того как получена верная Z-передаточная функция, необходимо запрограммировать её одним из указанных преподавателем способов (прямое, параллельное или последовательное программирование). Записать в отчет полученную структурную схему.

- Записать систему разностных уравнений и выписать главную матрицу A , матрицу-столбец B , матрицу-строку C и матрицу D .
- Включить тумблер +27 В на панели блока «Усилитель, генератор, демодулятор».
- Вызвать окно «Настройки», выбрать закладку «Сигнал» и установить такт квантования 5 мс, амплитуду сигнала 50 мрад и 120 мрад. На закладке «Регулятор» выбрать тип регулятора «Цифровой фильтр», установить полученный коэф. передачи и ввести матрицы A , B , C и D . Зафиксировать изменения и запустить управление. Занести результаты в отчет.
- Вызвать окно «Настройки», выбрать закладку «Сигнал» и установить амплитуду сигнала 50 мрад и 120 мрад. На закладке «Регулятор» выбрать тип регулятора «П-регулятор». Зафиксировать изменения и запустить управление. Занести результаты в отчет.
- Выключить двигатель. Закрыть программу. Выключить блок питания привода.

4. Содержание отчета

- Название и номер работы, цель работы;
- ЛАЧХ исходной и скорректированной системы, а также ЛАЧХ фильтра.
- Коэффициент передачи фильтра, передаточная функция аналогового фильтра, Z -передаточная функция цифрового фильтра,
- Структурная схема Z -передаточной функции цифрового фильтра
- Система разностных уравнений и матрицы A , B , C , D .
- Все переходные процессы с показателями качества, указанные в порядке выполнения работы;
- Выводы.

5. Контрольные вопросы

- Как по ЛАЧХ определить коэффициент передачи по контуру?
- Какой наклон должна иметь ЛАЧХ в районе частоты среза?
- Какой способ программирования наиболее выгоден? Почему?
- Почему матрица B в данной системе является матрицей столбцом, а не прямоугольной матрицей? Что такое матрица D ?

VIII. БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Горячев О.В. Компьютерное управление мехатронными системами. Часть 2: учеб.пособие (практикум). Тула: Изд-во ТулГУ, 2014.152с.
2. "Буран". Основы проектирования интеллектуальной системы управления орбитальным кораблем на атмосферном участке полета: монография / В. П. Андреев [и др.] ; под ред. А. С. Сырова; Федер. космич. агентство, Моск. опытно-конструктор. бюро «Марс» .— Москва : Марс, 2013 .— 276 с. : ил. — Дар каф. САУ ТулГУ ТулГУ : 1348233-1348246 .— Библиогр. в конце кн. — ISBN 978-5-9903829-3-0 (в пер.) .
3. Макаров Н.Н., Семашкин В.Е. Численные методы в анализе и синтезе систем управления: учеб. Пособие. Тула: Изд-во ТулГУ, 2014. 121с.
4. Дьяконов В. MATLAB: учебный курс. – СПб: Питер, 2001. – 560 с.: ил.
5. Цифровая обработка сигналов / А.Б. Сергиенко – СПб.: Питер, 2003. – 604 с.: ил.
6. Руководство по системе Matlab.