

Мультиметры APRA
 Выработка до 0,05%
 Работают под АСМ-протоколом
 с разрешением 10 бит и 3 1/2

Поиск: [] ОК [] Документация [] Магазины [] Новости [] Литература [] Схемы [] Конференции [] Реклама

I.ZMFC.RU: наш новый магазин. Лучшие цены на приборы!

Архив номеров: #9 2001г

ChipNews: [] О проекте [] В работе [] Оформление [] Подписки [] Реклама в журнале [] Рубрикатор [] Ссылки

Журналы: [] Chip News [] Радио

Хит ChipNews: [] Использование конденсаторов в датчиках энергии

MatLab для DSP. Расчет цифровых фильтров с учетом эффектов квантования

В. Анохин, А. Лантэ

MatLab для DSP. Расчет цифровых фильтров с учетом эффектов квантования

Разработка цифровых фильтров, сполни-зующих в своей работе пакет MATLAB, несомненно, смогл оценить многочисленые достоинства интерактивной графической программы (GUI) sptool [1]. Тем не менее, результаты, получаемые с помощью sptool, не всегда достаточны для специалистов, занимающихся проектированием и реализацией цифровых фильтров на базе специализированных, в частности, сигнальных, процессоров (DSP).

Характеристики фильтра, рассчитанно го на PC и реализованного на DSP, могут существенно отличаться. Это связано с тем, что разные цифровые системы (в том числе и персональный компьютер, на котором выполнялся расчет фильтра) используют различные типы арифметики (с плавающей или фиксированной точкой), имеют различные форматы и, как следствие, различный динамический диапазон и различную точность представления чисел. В итоге параметры фильтров, отчеты входных и выходных сигналов представляются лишь значениями из конечного набора чисел, то есть являются квантованными по уровню. Эффекты квантования отражаются также и на результатах математических операций, выполняемых сумматорами и умножителями. Из сказанного следует, что процесс проектирования должен включать процедуру коррекции полученных результатов с учетом эффектов квантования применительно к тем процессорам, на которых предполагается реализовывать синтезируемые фильтры. Эти процессоры, а также вычислительные системы, построенные на их основе, будем называть целевыми (в англоязычной литературе - Target Processors, Target Computers).

В последней версии MATLAB 6 (Release 12) появилась новая библиотека (toolbox) Filter Design, предназначенная для решения такого рода задач. Функции этой библиотеки дают возможность работать с тремя типами объектов - квантователями сигналов, квантованными фильтрами и квантованным преобразованием Фурье, используя разные форматы представления чисел. Наряду с функциями, позволяющими разрабатывать собственные программы, библиотека имеет интерактивную графическую программу (GUI) fdatool, с помощью которой можно рассчитывать выходные фильтры с учетом эффектов квантования, а также выбирать стандартные и определяемые пользователем форматы представления чисел. Помимо этого, fdatool предоставляет многочисленные варианты реализационных схем, что дает возможность оптимизировать решения по совокупности таких параметров, как динамический диапазон, шум арифметики, объем вычислений на отчет, предельные циклы. И хотя методы такой оптимизации в данном GUI не рассмотрены, представленные многочисленные структуры фильтра позволяют достаточно обсуждать подобные задачи.

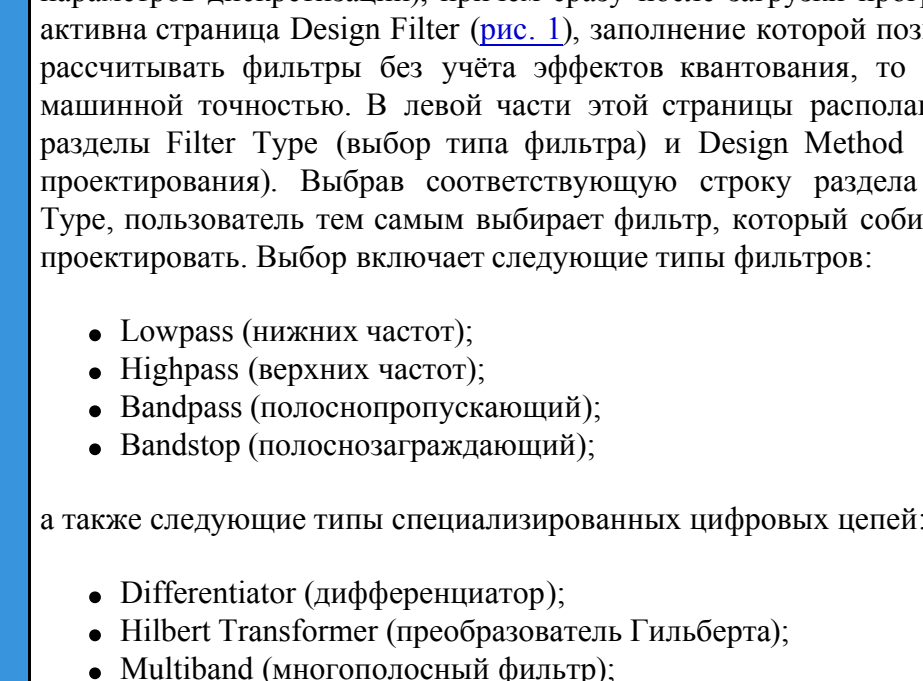
Материал настоящей статьи излагается в следующей последовательности. Во втором разделе дается описание основных объектов и органов управления главного окна программы fdatool. В третьем формулируется задача проектирования цифрового фильтра, в процессе решения которой излагается дальнейший материал. Четвертый и пятый разделы содержат информацию о форматах представления чисел и особенностях выполнения операций при использовании арифметики с фиксированной и плавающей точкой. В шестом разделе показаны некоторые приемы работы и возможности программы fdatool по расчету цифровых фильтров с учетом эффектов квантования.

Описание главного окна программы

Главное окно программы показано на рис. 1, оно появляется на экране после загрузки программы, для чего надо набрать её имя:

```
>> fdatool.
```

Рисунок 1. Главное окно программы fdatool



Как видно из рисунка, главное окно включает несколько областей, или разделов для ввода и вывода необходимой информации. В левом верхнем углу находится раздел Current Filter Information (информация о фильтре), где отображается информация о фильтре, с которым в текущий момент выполняется работа, а именно:

- форма реализации;
- источник (получен ли фильтр с помощью fdatool или импортирован из другого приложения);
- устойчивость;
- количество звеньев.

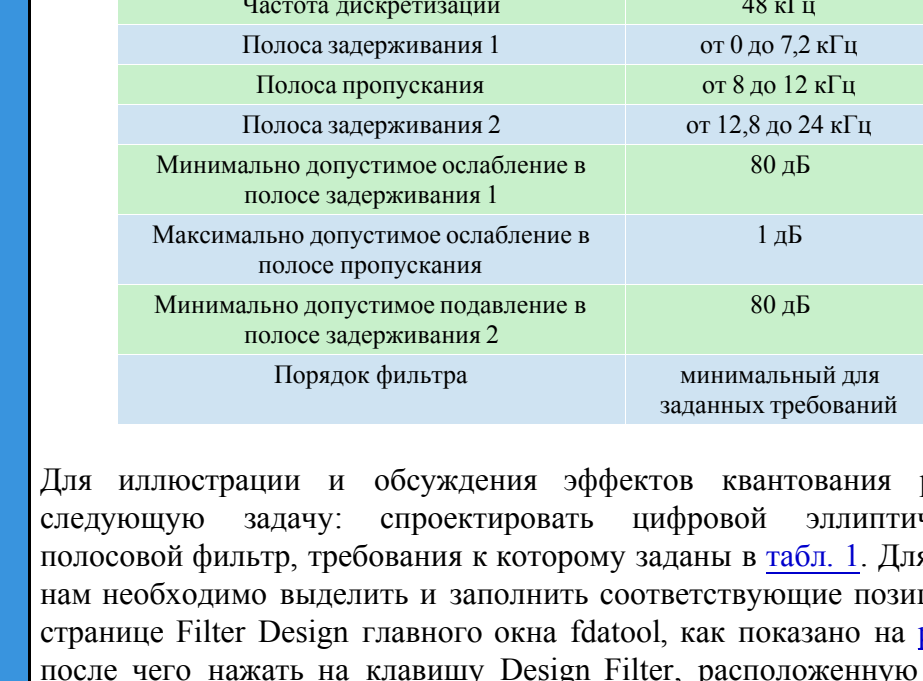
В дополнение к перечисленному, в этом разделе находится кнопка Convert Structure (преобразовать структуру), с помощью которой можно выбрать форму (схему) реализации фильтра.

Правее раздела Current Filter Information находится область, предназначенная для графического и численного отображения следующих характеристик и параметров фильтров:

- Filter Specifications (спецификация);
- Magnitude Response (амплитудно-частотная характеристика, или АЧХ);
- Phase Response (фазо-частотная характеристика, или ФЧХ);
- Magnitude and Phase Response (АЧХ и ФЧХ);
- Group Delay (групповое время задержки);
- Impulse Response (импульсная характеристика);
- Step Response (переходная характеристика);
- Pole/Zero Plot (полосы и нули);
- Filter Coefficients (коэффициенты фильтра).

Сразу после загрузки программы в этой области, как показано на рис. 1, отображается окно спецификации фильтра. Для отображения того или иного окна нужно нажать на одну из кнопок, расположенных сверху главного окна под строкой меню. Эти кнопки, а также их назначения показаны на рис. 2 в виде увеличенного фрагмента главного окна. Кроме отмеченных кнопок, те же действия можно выполнить, выбирая соответствующие пункты меню Analysis.

Рисунок 2. Управляющие кнопки для отображения характеристик и параметров фильтров



Нижняя половина главного окна содержит две страницы: Design Filter (проектирование фильтра) и Set Quantization Parameters (установка параметров дискретизации), причём сразу после загрузки программы активна страница Design Filter (рис. 1), заполнение которой позволяет рассчитывать фильтры без учёта эффектов квантования, то есть с машинной точностью. В левой части этой страницы располагаются разделы Filter Type (выбор типа фильтра) и Design Method (метод проектирования). Выбрав соответствующую строку раздела Filter Type, пользователь тем самым выбирает фильтр, который собирается проектировать. Выбор включает следующие типы фильтров:

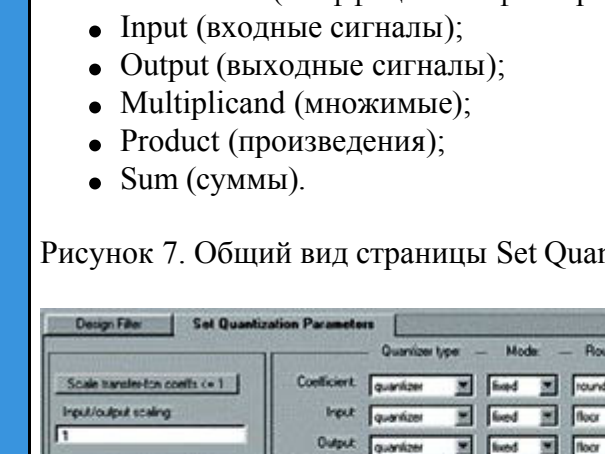
- Lowpass (нижних частот);
- Highpass (верхних частот);
- Bandpass (полоснопропускающий);
- Bandstop (полосозаграждающий);

а также следующие типы специализированных цифровых цепей:

- Differentiator (дифференциатор);
- Hilbert Transform (преобразователь Гильберта);
- Multiband (многополосный фильтр);
- Arbitrary Magnitude (фильтр с произвольной АЧХ, форма которой определяется пользователем);
- Arbitrary Group Delay (фильтр с произвольным групповым временем задержки).

Раздел Design Method позволяет сделать выбор между фильтрами с бесконечными и конечными импульсными характеристиками (IR - Infinite Impulse Response и FIR - Finite Impulse Response, соответственно). В случае выбора IIR-фильтра, необходимо конкретизировать вид прототипа фильтра: Баттерворта, Чебышева 1-го рода, Чебышева 2-го рода и эллиптический (рис. 3а). Если же выбраны FIR-фильтры, выбор включает расчёт равнополосных фильтров, а также расчёт методом наименьших квадратов и оконным методом (рис. 3б). Если проектирование выполняется оконным методом, активизируется область Window Specifications (спецификация окна), расширяющаяся рядом с областью Design Method, в которой имеется раскрывающееся меню Window с набором спектральных окон.

Рисунок 3. Выбор метода расчёта при проектировании фильтров с бесконечными (а) и конечными (б) импульсными характеристиками



Правую часть страницы Filter Design занимают разделы Frequency Specifications and Magnitude Specifications (частотные и амплитудные спецификации, соответственно). Первый из них содержит редактируемые окна для ввода значений частот дискретизации Fz и граничных частот полос пропускания Fs и задерживания Fstop, а также единиц измерения АЧХ - Units, причём количество граничных частот зависит от типа фильтра, задаваемого в разделе Filter Type. Второй раздел - Magnitude Specifications - позволяет задать требования амплитудной характеристики проектируемого фильтра для областей пропускания (Apass или Wpass) и задерживания (Astop или Wstop). Оба обсуждаемых раздела имеют раскрывающиеся меню Units для выбора единиц измерения частоты и амплитуды. На странице Filter Design имеется ещё одна область, а именно Filter Order (порядок фильтра), в которой можно указать явно порядок проектируемого фильтра, либо потребовать, чтобы программа автоматически выбрала наименьший порядок, в соответствии с введённой спецификацией.

Теперь, когда известны назначения разделов страницы Design Filter, можно было бы приступить к описанию второй страницы - Set Quantization Parameters, однако анализировать эффекты квантования удобнее и нагляднее, если уже имеется спроектированный цифровой фильтр, коэффициенты которого рассчитаны с машинной точностью. Придерживаясь терминологии, используемой в справочной системе MATLAB, цифровой фильтр, коэффициенты которого рассчитаны с машинной точностью, будем называть цифровым фильтром-прототипом (в оригинале - Referenced Filter).

Проектирование цифрового фильтра: постановка задачи и квантование параметров

Рисунок 4. Страница Design Filter, содержащая данные табл. 1

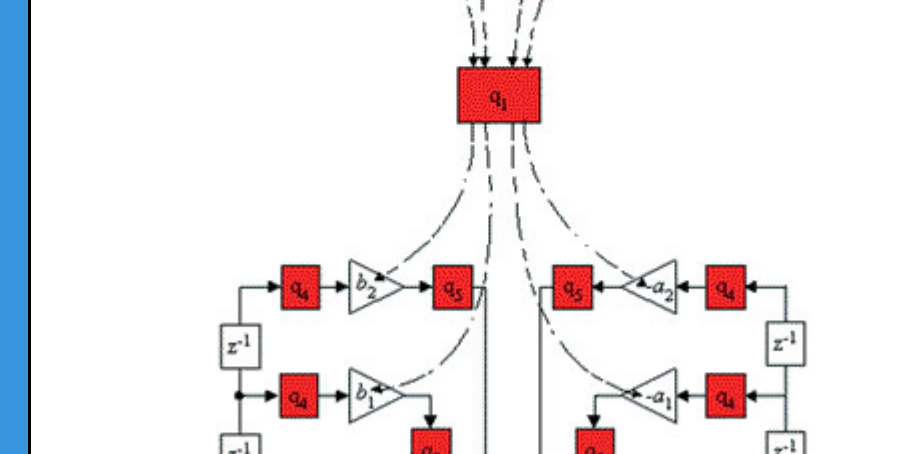
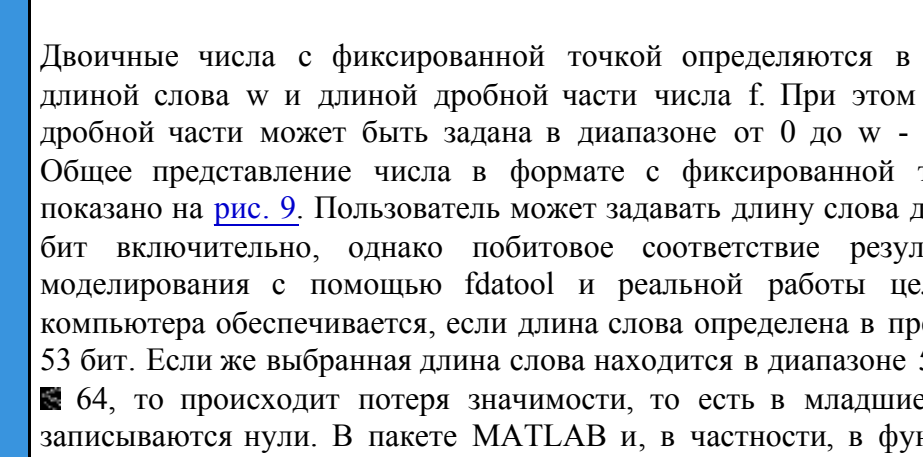


Таблица 1

Частота дискретизации	48 кГц
Полоса задерживания 1	от 0 до 12 кГц
Полоса пропускания	от 8 до 7,2 кГц
Полоса задерживания 2	от 12,8 до 24 кГц
Минимально допустимое ослабление в полосе задерживания 1	80 дБ
Максимально допустимое ослабление в полосе пропускания	1 дБ
Минимально допустимое ослабление в полосе задерживания 2	80 дБ
Порядок фильтра	минимальный для заданных требований

Для иллюстрации и обсуждения эффектов квантования решим следующую задачу: спроектировать цифровой эллиптический полосовой фильтр, требования к которому заданы в табл. 1. Для этого нам необходимо выделить и заполнить соответствующие позиции на странице Filter Design главного окна fdatool, как показано на рис. 4, после чего нажать на клавишу Design Filter, расположенную внизу окна. После завершения расчёта в верхней части окна будут представлены результаты (рис. 5). Как видно из рисунка, на экране отображены характеристики и параметры выводится АЧХ полученного фильтра, однако пользователь может, по мере необходимости, для вывода перекрывать её содержанием, например, импульсную характеристику, полюсов и нулей и так далее.

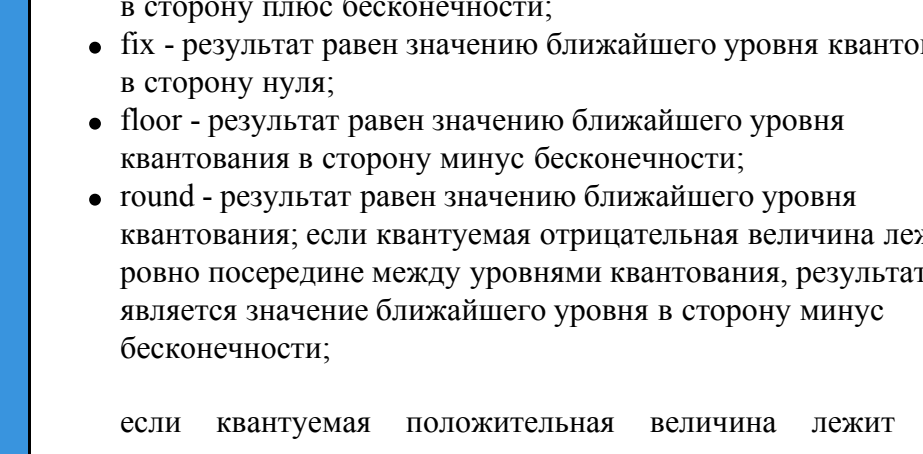
Рисунок 5. Амплитудно-частотная характеристика фильтра-прототипа



До настоящего момента все операции над данными числами выполнялись с машинной точностью, когда для представления чисел использовался формат double (в соответствии с этим форматом для записи и хранения числа отводится 8 байт). Теперь проанализируем, что произойдёт при изменении формата представления числа. Такая потребность возникает, когда необходимо выполнить эмюляцию работы построенного фильтра на базе цифровых процессоров, использующих отличные от PC форматы данных. В главном окне, ниже раздела Current Filter Information, находится раздел Quantization, содержащий единственное поле Turn Quantization On. Для запуска процедуры расчёта квантованного фильтра с параметрами квантования, заданными по умолчанию. (Также по умолчанию расчёт выполняется для прямой реализации фильтра.)

По окончании процедуры расчёта, АЧХ нового (квантованного) фильтра наносится на АЧХ фильтра-прототипа, рассчитанного ранее. Область отображения характеристик будет выглядеть так, как показано на рис. 6, где АЧХ фильтра-прототипа обозначена как Quantized, с учётом эффектов квантования (Quantized), существенно отличается от соответствующей характеристики фильтра-прототипа (Referenced). Сохраним полученные результаты на диске, для чего откроем меню File и выберем раздел Save Session As...

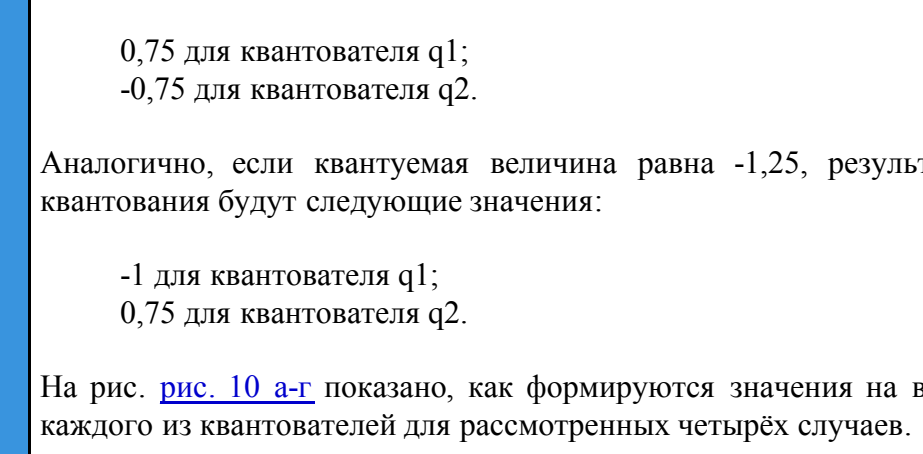
Рисунок 6. Амплитудно-частотные характеристики фильтра-прототипа и квантованного фильтра



Чтобы разобраться в причинах, приведших к таким сильным изменениям, перейдём на страницу Set Quantization Parameters, показанную на рис. 7, и проанализируем её содержание. Как следует из рисунка, имеется пять видов объектов, которые подвергаются квантованию:

- Coefficient (коэффициенты фильтра);
- Input (входные сигналы);
- Output (выходные сигналы);
- Product (произведения);
- Sum (суммы).

Рисунок 7. Общий вид страницы Set Quantization Parameters



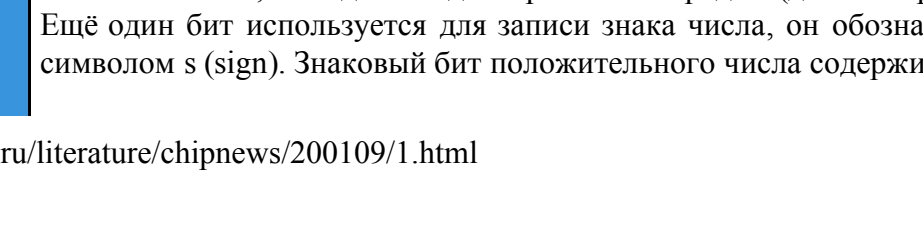
Для выполнения операции квантования используются квантователи определенных объектов. Таким образом, квантованный фильтр представляет собой нелинейную цифровую систему, включающую:

- квантователь входного сигнала;
- собственное filtro с квантованными значениями коэффициентов, квантователями множителей (сигналов на входах умножителей), произведений (сигналов на выходах умножителей) и сумм (сигналов на выходах сумматоров);
- квантователь выходного сигнала.

Пример квантованного фильтра 2-го порядка показан на рис. 8, где указаны следующие обозначения:

- q1 - квантователь коэффициентов;
- q2 - квантователь входного сигнала;
- q3 - квантователь выходного сигнала;
- q4 - квантователь множителя;
- q5 - квантователь произведения;
- q6 - квантователь суммы.

Рисунок 8. Структурные схемы фильтра-прототипа и квантованного фильтра 2-го порядка



Свойства квантованного фильтра в целом зависят от параметров каждого из перечисленных квантователей. Эти параметры определяются содержанием пяти колонок, или полей, расположенных правее имён объектов квантования (рис. 7). Параметры определяют, каким образом формируется сигнал или вычисляется коэффициент на выходе соответствующего квантователя.

Свойства квантованного фильтра зависят также от структуры, или формы реализации фильтра. В левом нижнем углу страницы Set Quantization Parameters имеется клавиша Show Filter structures... нажав на которую пользователь получит возможность просмотреть различные формы реализации фильтров. Подчеркнем, что появляющиеся схемы никак не связаны с тем фильтром, который проектируется в момент просмотра. Их можно интерпретировать как своего рода справочник или альбом, помогающий необходимо в главном окне нажать кнопку Convert Structure, в появившемся окне выделить желаемую структуру и нажать ОК.

Говоря о квантователях, прежде всего подчеркнём тот очевидный факт, что реализация любого цифрового фильтра основывается на использовании арифметики с фиксированной или плавающей точкой (Fixed-Point или Floating-Point Arithmetic, соответственно). В связи с этим, остановимся кратко на форматах представления чисел и особенностях выполнения арифметических операций в программе fdatool.

Арифметика с фиксированной точкой: форматы данных и реализация операций

Выбор формата каждого из квантователей зависит от фиксированной или плавающей точкой задается значениями параметров колонки Mode, расположенной на странице Set Quantization Parameters. Для реализации арифметики с фиксированной точкой необходимо выбрать значение fixed.

Двоичные числа с фиксированной точкой определяются в битах двоичного слова w и длиной дробной части числа f. При этом длина дробной части может быть задана в диапазоне от 0 до w - 1 бит. Общее представление числа в формате с фиксированной точкой показано на рис. 9. Пользователь может задавать длину слова до 64-х бит включительно, однако побитовое соответствие результатов моделирования с помощью fdatool и реальной работы целевого компьютера обеспечивается, если длина слова определена в пределах 53 бит. Если же выбранная длина слова находится в диапазоне 54 и w 64, то происходит потеря значимости, то есть в младшие биты записываются нули. В пакете MATLAB и, в частности, в функциях библиотеки Filter Design, формат для чисел с фиксированной точкой задётся в виде [w, f] в колонке format.

Рисунок 9. Общее представление числа в формате с фиксированной точкой

Числа с фиксированной точкой могут быть беззнаковыми или со знаком. В первом случае старший бит, как и остальные, используется для представления величины числа, тогда как во втором - для его знака (0 соответствует знаку "плюс", 1 - знаку "минус"). Динамический диапазон для беззнаковых чисел равен $[0, 2^w - 2^f]$, а для чисел со знаком $[-2^{w-f-1}, 2^w - f - 2^f]$. В обоих случаях точность, то есть разность двух ближайших чисел в данном диапазоне, равна $\square = 2^f$.

В связи с тем, что результатом квантования является замена квантуемой величины числом из конечного, предопределённого набора чисел, называемых уровнями квантования, необходимо ввести информацию о том, каким образом эту замену использовать (значения и число уровней квантования определяются используемым форматом). Для этого в программе fdatool имеется поле Round mode, где для каждого квантователя можно выбрать способ округления. Пользователь имеет возможность выбрать один из следующих режимов:

- ceil - результат равен значению ближайшего уровня квантования в сторону плюс бесконечности;
- fix - результат равен значению ближайшего уровня квантования в сторону нуля;
- floor - результат равен значению ближайшего уровня квантования в сторону минус бесконечности;
- round - результат равен значению ближайшего уровня квантования; если квантуемая отрицательная величина результат равно середине между уровнями квантования, результат является значением ближайшего уровня в сторону минус бесконечности;

если квантуемая положительная величина лежит ровно посередине между уровнями квантования, результатом является значение ближайшего уровня квантования в сторону плюс бесконечности.

- convergent - способ округления такой же, как и round, однако, если квантуемая величина лежит ровно посередине между уровнями квантования, то округление по правилам round выполняется лишь в том случае, когда в младший бит после округления записывается единица.

Когда квантованию подвергается величина, значение которой находится за пределами динамического диапазона квантователя, возникает явление переполнения. Способ обработки таких величин задётся в поле Overflow Mode (режим переполнения) для каждого квантователя. Пользователь может выбрать режим saturate или wrap.

Режим saturate означает разрешение работы с насыщенем: квантуемая величина, лежащая вне динамического диапазона, полагается равной ближайшему предельно допустимому числу. В случае выбора режима wrap, старшие биты квантуемой величины, лежащие вне динамического диапазона, будут отброшены, и результат квантования будет представлять младшими w-битами, при этом для чисел со знаком старший бит b_{w-1} (рис. 9).

Рассмотрим следующий пример. Пусть имеются два квантователя - q1 и q2, для которых задан формат fixed [3 2], и пусть первый из них работает в режиме saturate, а второй - в режиме wrap. В этом случае шаг квантования для обоих квантователей $\square = 0,25$, динамический диапазон $[-1, 1, 0,75]$. Если входная квантуемая величина равна 1,25, то на выходах этих квантователей будем иметь:

- 0,75 для квантователя q1;
- -0,75 для квантователя q2.

Аналогично, если квантуемая величина равна -1,25, результатами квантования будут следующие значения:

- -1 для квантователя q1;
- 0,75 для квантователя q2.

На рис. рис. 10 а-г показано, как формируются значения на выходе каждого из квантователей для рассмотренных четырех случаев.

Рисунок 10. Варианты квантования в режиме переполнения: положительного числа квантователем q1 (а), отрицательного числа квантователем q2 (б) и отрицательного числа квантователем q2 (г)

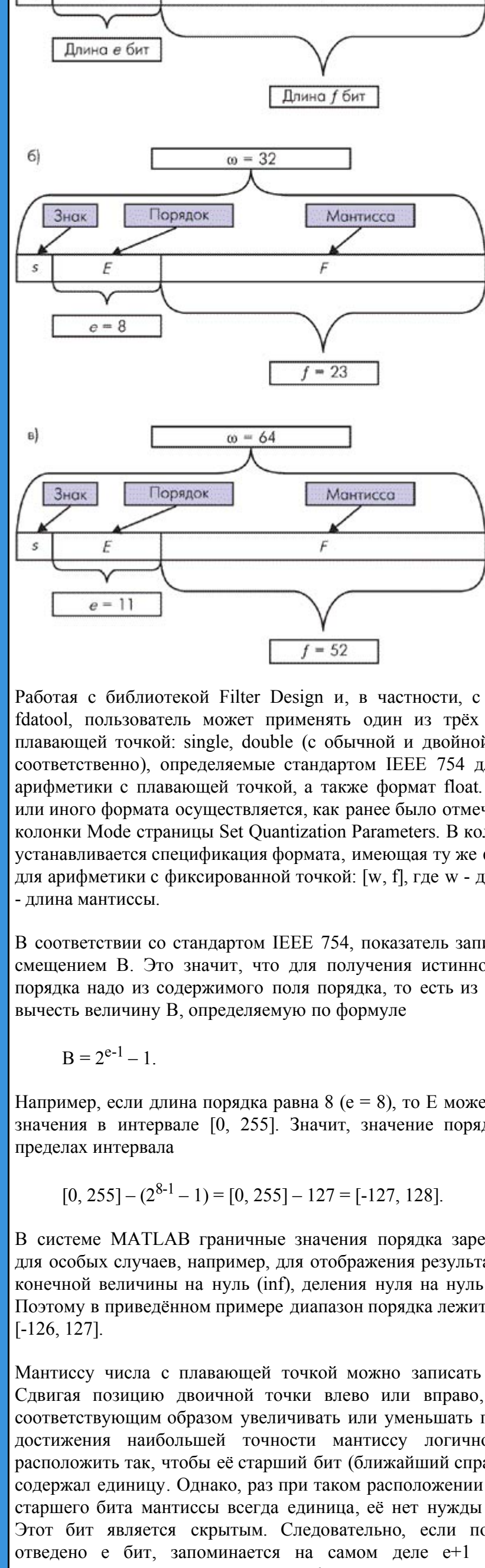
Функции библиотеки Filter Designer позволяют использовать два типа квантователей: quantizer и unquantizer. Они работают практически одинаково, за исключением того, что на выходе квантователя unquantizer будет единица, если на его вход поступит величина из диапазона $[1 - \square, 1 - \square]$. Например, пусть тип первого из рассмотренных нами видов квантователей (q1) quantizer, а второго (q2) - unquantizer. Если на вход q1 подать единицу, то возникнет переполнение, так как используется формат fixed [3 2], и на выходе будет значение 0,75 (максимальная величина динамического диапазона квантователя q1). Если же единицу подать на вход квантователя q2, переполнения не возникнет, а на его выходе будет единица. В fdatool выбор типа квантователя осуществляется на странице Set Quantization Parameters в колонке Quantizer Type.

Арифметика с плавающей точкой: форматы данных и реализация операций

Недостаток описания чисел в формате с фиксированной точкой является неудобство представления очень больших и очень маленьких чисел при использовании разумной длины слова w. Это ограничение снимается, если используется формат описания с плавающей точкой. Любое двоичное число с плавающей точкой можно представить в виде $\pm E \cdot 2^E$, где E обозначает мантиссу, или дробную часть, 2 - основание системы исчисления, а E - порядок. Длина слова при представлении числа в формате с плавающей точкой $w = f + E + 1$, где f - количество бит, отводимых для хранения мантиссы (длина мантиссы), а e - количество бит, отводимых для хранения порядка (длина порядка). Ещё один бит используется для записи знака числа, он обозначается символом s (sign). Знаковый бит положительного числа содержит нуль

(s = 0), отрицательного - единицу (s = 1). Общее представление числа в формате с плавающей точкой показано на [рис. 11 а](#).

Рисунок 11. Представление числа в формате с плавающей точкой: общее представление (а), а также форматы (б) и double (в)



Работая с библиотекой Filter Design и, в частности, с программой fdatool, пользователь может применить один из трёх форматов с плавающей точкой: single, double (с обычной и двойной точностью соответственно), определённые стандартом IEEE 754 для двойной арифметики с плавающей точкой, а также формат float. Выбор того или иного формата осуществляется, как ранее было отмечено, в окне колонки Mode страницы Set Quantization Parameters. В колонке Format устанавливается спецификация формата, имеющая ту же форму, что и для арифметики с фиксированной точкой: [w, f], где w - длина слова, f - длина мантиссы.

В соответствии со стандартом IEEE 754, показатель записывается со смещением В. Это значит, что для получения истинного значения порядка надо из содержимого поля порядка, то есть из величины Е, вычесть величину В, определяемую по формуле

$$B = 2^{e-1} - 1.$$

Например, если длина порядка равна 8 (e = 8), то Е может принимать значения в интервале [0, 255]. Значит, значение порядка лежит в пределах интервала

$$[0, 255] - (2^{8-1} - 1) = [0, 255] - 127 = [-127, 128].$$

В системе MATLAB граничные значения порядка резервируются для особых случаев, например, для отображения результатов деления конечной величины на нуль (inf), деления нуля на нуль (NaN) и др. Поэтому в приведённом примере диапазон порядка лежит в интервале [-126, 127].

Мантисса числа с плавающей точкой можно записать по-разному. Сдвигая позицию двоичной точки влево или вправо, надо лишь соответствующим образом увеличивать или уменьшать порядок. Для достижения наибольшей точности мантиссы логично было бы расположить так, чтобы её старший бит (ближайший справа от точки) содержал единицу. Однако, раз при таком расположении содержимое старшего бита мантиссы всегда единица, её нет нужды запоминать. Этот бит является скрытым. Следовательно, если до бит мантиссы отведено e бит, запоминается на самом деле e+1 бит (е бит отведено за старшим, скрытым битом мантиссы, плюс один скрытый бит). Таким образом, взаимосвязь между величиной, записанной в формате, показанном на [рис. 11 а](#), и истинным значением у задаётся в виде

$$v = (-1)^s 2^{(E-B)} (1.F).$$

Числа, представленные таким способом, называются нормализованными.

В некоторых случаях может возникнуть ситуация, когда результатом какой-либо арифметической операции является число, имеющее такое маленькое абсолютное значение, что для него величина

$$E = \text{порядок} + B < 0.$$

Это явление называется исчезновением разрядов порядка, или антипереносом порядка. Для того, чтобы увеличить порядок и тем самым избежать явления переполнения, используются денормализованные числа. Они получаются из соответствующего представления сдвигом мантиссы вправо и нормализованном увеличением порядка (то есть уменьшением модуля порядка). Для денормализованных чисел взаимосвязь записи в формате, показанном на [рис. 11 а](#), и истинного значения определяется соотношением

$$v = (-1)^s 2^{-(B+1)} (0.F).$$

Читатель может заметить, что для задания знака числа с плавающей точкой отведён один бит, в то время как знак имеет и мантисса, и порядок. В описываемых форматах знаковый бит содержит знак мантиссы, а знак же порядка определяется по результату вычитания смещения В из содержимого поля порядка Е.

Представления чисел при использовании форматов single и double показаны на [рис. 11 б](#) и в, соответственно. При выборе одного из этих форматов автоматически устанавливаются параметры в колонках Round mode, Overflow mode и Format. Параметрами Round mode в этом случае являются значения round, а Overflow mode - saturate. В [табл. 2](#) указаны допустимые значения порядков и численных значений при использовании форматов single и double.

Таблица 2. Допустимые значения порядков и численных значений при использовании форматов single и double

Формат	Порядок	Single	Значение	Порядок	double	Значение
Нормализованные	0 < E < 255	$v = (-1)^s 2^{E-127} (1.F)$		0 < E < 2047	$v = (-1)^s 2^{E-1023} (1.F)$	
Денормализованные	E = 0	$v = (-1)^s 2^{-(126-F)} (0.F)$		E = 0	$v = (-1)^s 2^{-(1022-F)} (0.F)$	

Ещё один формат, с которым работает fdatool - это float. Представление чисел в этом формате также соответствует приведённым формулам и обозначениям, показанным на [рис. 11 а](#), но в этом случае пользователь может произвольно задавать длину мантиссы и порядка, а также устанавливать значения в полях Format и Round mode. При этом длина порядка должна находиться в интервале $1 \leq e \leq 11$, а длина слова - в интервале $e+1 \leq w \leq 64$. Независимо от выбранного формата (single, double или float), в случае возникновения переполнения на выходе соответствующего квантователя устанавливается значение inf или -inf, поэтому окна колонки Overflow mode неактивны.

Рисунок 12. Параметры фильтра-прототипа и квантованного фильтра

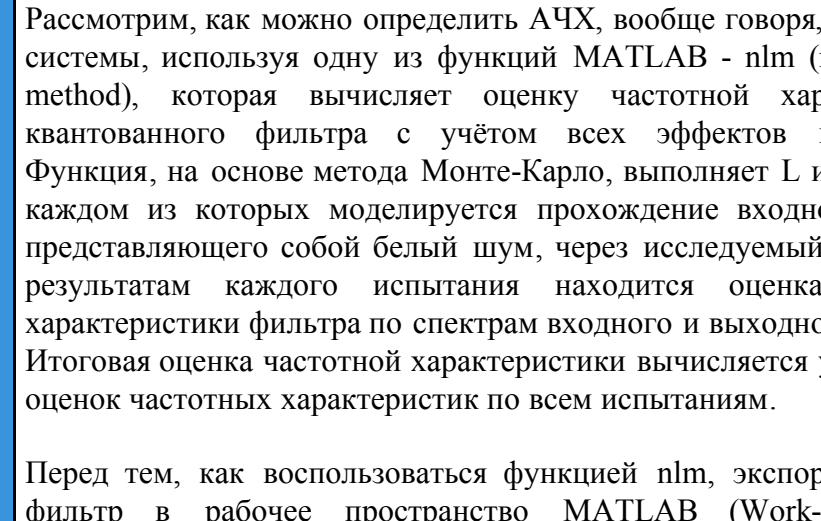
```
Quantized Direct form II transposed filter
Numerator
QuantizedCoefficients{1} ReferenceCoefficients{1}
(1) 0.000335693359375 0.00033031554448794
(2) -0.001007080078125 -0.002580261271031570
(3) 0.002593994140625 0.002580261271031570
(4) -0.0045776371875 0.004580740133027181
(5) 0.007554736326125 0.007347272737754056
(6) -0.009704589843750 -0.009715041792670265
(7) 0.011993408203125 0.011986955602516003
(8) -0.01212558593750 -0.013120022874790487
(9) 0.01293944532500 0.0130304093743245
(10) -0.01312558593750 -0.013120022874790454
(11) 0.011993408203125 0.011986955602515940
(12) -0.009704589843750 -0.00971504179267119
(13) 0.007554736326125 0.007347272737753959
(14) -0.0045776371875 -0.004580740133027142
(15) 0.002593994140625 0.002580261271031547
(16) -0.001007080078125 -0.001020081231375187
(17) 0.000335693359375 0.000330315544487902
Denominator
QuantizedCoefficients{2} ReferenceCoefficients{2}
(-2) -1.000000000000000 -4.0018106715603197
(-3) 0.999998482421875 13.979597370763326
(-4) -1.000000000000000 -31.703692238145337
(-5) 0.999998482421875 63.654203987427906
(-6) -1.000000000000000 -100.2143446666917
(-7) 0.999998482421875 141.701891018442469
(-8) -1.000000000000000 -168.0189747922654
(-9) 0.999998482421875 175.86734947768485
(-10) -1.000000000000000 -156.48177861802839
(-11) 0.999998482421875 128.8638778866985
(-12) 0.000000000000000 -53.8325505073258
(-13) 0.999998482421875 50.20128500716024
(-14) -1.000000000000000 -23.53851009548396
(-15) 0.999998482421875 9.774298471347578
(-16) -0.000000000000000 -2.6300444174959825
(-17) 0.619323730468750 0.619323730468750
FilterStructure = dI2
ScaleValues = [1]
NumberOfSections = 1
StatesPerSection = 169
CoefficientFormat = quantizer'fixed', 'round', 'saturate', [16 15]
InputFormat = quantizer'fixed', 'floor', 'saturate', [16 15]
OutputFormat = quantizer'fixed', 'floor', 'saturate', [16 15]
MultiplierFormat = quantizer'fixed', 'floor', 'saturate', [16 15]
ProductFormat = quantizer'fixed', 'floor', 'saturate', [32 30]
SumFormat = quantizer'fixed', 'floor', 'saturate', [32 30]
Warning: 16 overflows in coefficients.
```

Проектирование цифрового фильтра: квантование параметров и анализ результатов

Теперь, после знакомства с основными причинами, порождающими эффекты квантования, вновь обратимся к нашей задаче проектирования цифрового фильтра. Ясно, что различие между АЧХ фильтра-прототипа и АЧХ квантованного фильтра ([рис. 6](#)) при выбранной структуре связано с установками параметров квантования. Как видно из [рис. 7](#), в данном случае использовались форматы с фиксированной точкой [16, 15] и [32, 30], что соответствует значению fixed в поле Mode и значениям [16, 15] для quantizer и multiplier. Важно обратить внимание на установленный формат для квантователя Coefficient равен [16, 15]. Это означает, что единственный бит, не предназначенный для хранения дробной части числа, является знаковым, то есть квантование любого числа, модуль которого больше или равен единице, приведёт к переполнению. Из-за эффекта переполнения коэффициенты знаменателя квантованного фильтра существенно отличаются от соответствующих коэффициентов фильтра-прототипа, что отмечено в начале каждой строки, за исключением последней, знаком "+" (переполнение в сторону плюс бесконечности) или "-" (переполнение в сторону минус бесконечности).

Попробуем промасштабировать коэффициенты так, чтобы они по модулю не превышали единицу. Это необходимо для повышения точности расчётов при реализации фильтров на DSP и осуществляется нажатием кнопки Scale transfer-fcn coeffs <= 1, расположенной в левой верхней части страницы Set Quantization Parameters. Для масштабирования используются коэффициенты, равные степени 2. После выполнения расчётов в поле Input/output scaling, расположенном под указанной кнопкой, появятся значения масштабных множителей, на которые умножаются входной и выходной сигналы фильтра с изменёнными коэффициентами. АЧХ фильтров будут выглядеть так, как показано на [рис. 13](#).

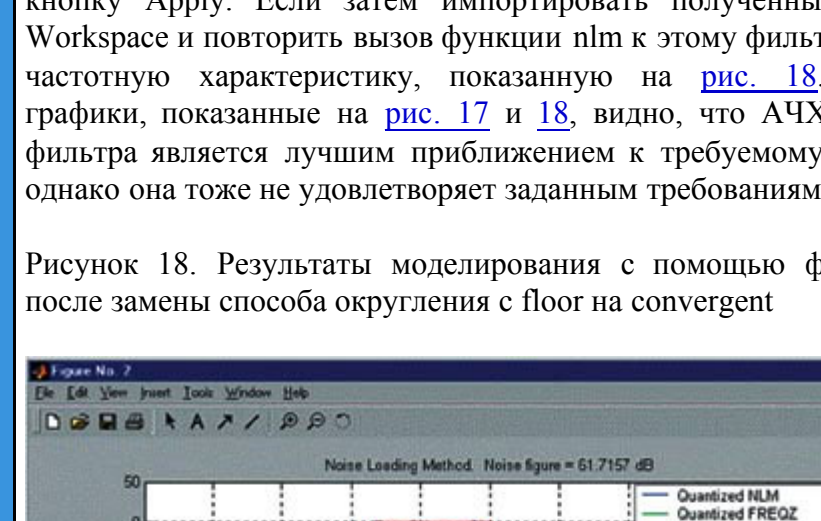
Рисунок 13. Амплитудно-частотные характеристики фильтров после масштабирования коэффициентов



Помимо искажений АЧХ, в результате квантования появляются шумы, фильтр может стать неустойчивым, и возникнуть разные виды переполнения (например, генерация периодических колебаний, превышения предела высокого и низкого уровней) [2], что в принципе затрудняет определение АЧХ. fdatool не позволяет решить задачи расчёта шумов арифметики, расчёта предельных значений, оптимизации динамического диапазона и ряд других. Для некоторых из этих целей в MATLAB имеются специальные функции (например, nlm, limitcycle и другие), что он позволяет создавать нелинейные модели цифровых фильтров, весьма точно отражающих динамику реальных систем. Эти модели и являются объектами для упомянутых выше функций.

Возвращаясь к нашему примеру, шёлком мышью, поместив предварительно курсор в редакторное окно Input/output scaling, нажмём на клавиатуре Enter, и затем нажмём клавишу Apply, находящуюся внизу главного окна fdatool. В области Current Filter Information появится сообщение "Stable: No". В том, что фильтр неустойчив, можно также убедиться, нажав на кнопку PoleZero Plot (на [рис. 2](#) она обозначена как "Полосы и нули"). В результате появится карта нулей и полюсов, показанная на [рис. 14](#), на которой видно, что передаточная функция построенного квантованного фильтра имеет полюсы, лежащие вне единичного круга.

Рисунок 14. Карта нулей и полюсов



Теперь попробуем решить поставленную задачу иначе, но прежде восстановим наши результаты, записанные на диск. Для этого выберем раздел Open Session меню File и откроем файл. В области Current Filter Information нажмём кнопку Convert Structure... и в открывшемся окне установим флажок Use second order sections ([рис. 15](#)), указав тем самым, что проектируем квантованный фильтр должен быть реализован звеньями второго порядка. Нажмите кнопки Apply или ОК инициализирует расчёт фильтра в указанной реализации. После вычисления можно убедиться, что АЧХ фильтра-прототипа и квантованного фильтра практически совпадают. Полученный результат показан на [рис. 16](#). Важно обратить внимание на содержимое окна Input/output scaling: в нём расчётные масштабные множители для сигналов на входе каждого из звеньев второго порядка, а также для выходного сигнала последнего звена.

Рисунок 15. Установка флажка Use second order sections в окне Convert Structure

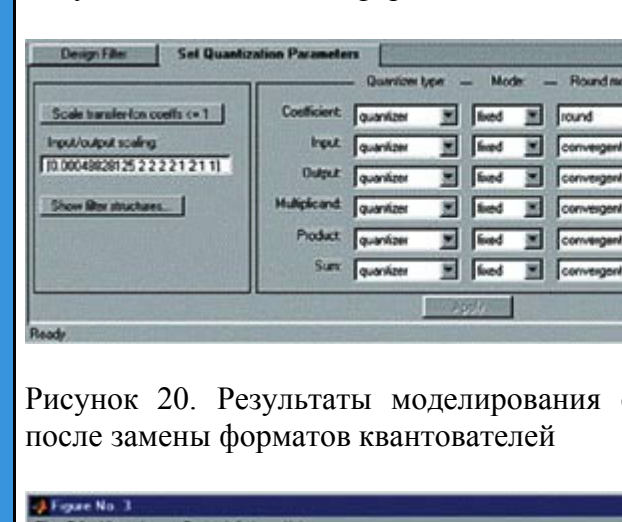
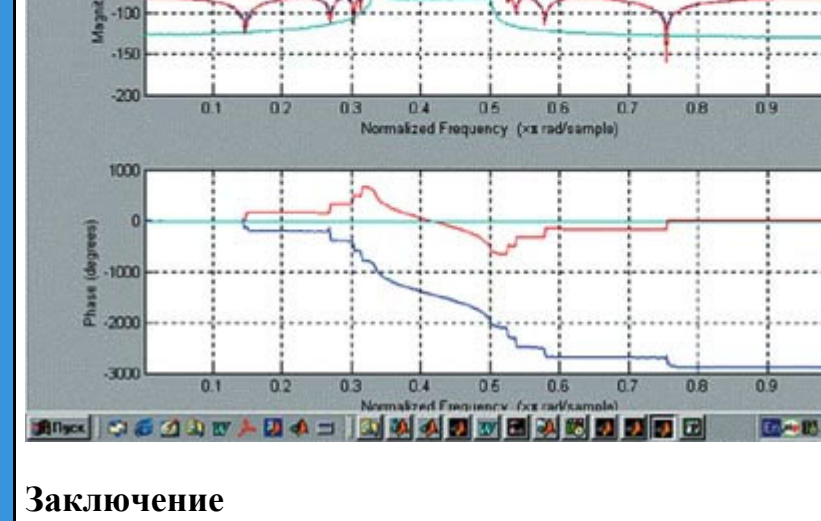


Рисунок 16. Амплитудно-частотные характеристики фильтра-прототипа и квантованного фильтра, реализованного звеньями второго порядка



Итак, на первый взгляд задача, поставленная в разделе 3 ([табл. 1](#)), решена - цифровой фильтр с требуемыми характеристиками построен. Однако следует заметить, что частотная характеристика фильтра, и, в частности, АЧХ, показанная на [рис. 16](#), построена по расчётным квантованным коэффициентам в предположении, что фильтр является идеальной линейной системой. Другими словами, при расчёте частотной характеристики учитывались лишь эффекты квантования коэффициентов фильтра. Для получения реальной частотной характеристики необходимо учесть структурные эффекты квантования, связанные с формой реализации (структурой) фильтра и квантованием других параметров, устанавливаемых на странице Set Quantization Parameters.

Чтобы получить частотную характеристику фильтра, построенную с учётом указанных факторов, следует выполнить комплексную проверку его работы на основе имитационного моделирования. Рассмотрим, как можно определить АЧХ, вообще говоря, нелинейной системы, используя одну из функций fdatool - nlm (non-linear loading method), которая вычисляет оценку частотной характеристики квантованного фильтра с учётом всех эффектов квантования. Функция, на основе метода Монте-Карло, выполняет L испытаний, в каждом из которых моделируется прохождение входного сигнала, представляющего собой белый шум, через исследуемый фильтр. По результатам каждого испытания находится оценка частотной характеристики фильтра по спектру входного и выходного сигналов. Итоговая оценка частотной характеристики вычисляется усреднением оценок частотных характеристик по всем испытаниям.

Перед тем, как воспользоваться функцией nlm, экспортируем наш фильтр в рабочее пространство MATLAB (Work-space). Это выполняется следующим образом. В меню File выбрать раздел Export, в появившемся окне указать, куда экспортировать (Work-space) и имя файла (например, Hq1), после чего нажать кнопку ОК. После выполнения команды надо перейти в командное окно MATLAB и набрать команду

```
>> nlm(Hq1,[],50),
```

по которой будет выполнено моделирование, включающее 50 испытаний. В результате появится окно с графиками АЧХ и ФЧХ, показанное на [рис. 17](#), из которых видно, что построенная нами нелинейная система, каковым является квантованный фильтр, не удовлетворяет заданным требованиям, изложенным в [табл. 1](#) и показанным на [рис. 4](#). Отметим, что функцию nlm можно вызвать, указав лишь имя фильтра:

```
>> nlm(Hq1).
```

Рисунок 17. Результаты моделирования с помощью функции nlm



В этом случае число испытаний L будет выбрано по умолчанию (L = 10), и получаемые графики будут менее сглаженными.

Очевидно, что для получения желаемого результата нам надо изменить параметры квантования, задаваемые на странице Set Quantization Parameters. Попробуем для всех квантователей, кроме квантователей коэффициентов, переуставить способ округления с floor в convergent и рассчитать новый квантованный фильтр, нажав кнопку Apply. Если затем импортировать полученный фильтр в Work-space и повторить вызов функции nlm и этому фильтру, получим частотную характеристику, показанную на [рис. 18](#). Сравнительная графика, показанная на [рис. 17](#) и [18](#), видно, что АЧХ последнего фильтра является лучшим приближением к требуемому результату, однако она тоже не удовлетворяет заданным требованиям.

Рисунок 18. Результаты моделирования с помощью функции nlm после замены способа округления с floor на convergent

Обратимся снова к странице Set Quantization Parameters и установим форматы квантователей в соответствии с [рис. 19](#). Повторим процедуру расчёта, импорта и испытания фильтра с помощью nlm, получим результат, показанный на [рис. 20](#). Как видно из рисунка, АЧХ фильтра-прототипа и квантованного фильтра практически совпадают, то есть полученный фильтр удовлетворяет требованиям, изложенным в [табл. 1](#).

Рисунок 19. Изменение форматов квантователей

Рисунок 20. Результаты моделирования с помощью функции nlm после замены форматов квантователей

Заключение

Материал, изложенный в настоящей статье, следует рассматривать как введение в GUI-fdatool, позволяющее решать ряд практически важных и теоретически интересных задач, связанных с проектированием цифровых фильтров. Главное - это возможность весьма точно моделировать разнообразную структуру цифровых фильтров в режиме нелинейных преобразований, связанных с эффектами квантования, и возможность оценивать интегральные параметры фильтров.

Вместе с тем, многие вопросы синтеза цифровых фильтров с оптимальными либо близкими к оптимальным характеристиками и учитывающие нелинейные эффекты квантования остаются за пределами этой публикации, равно как и за пределами системы MATLAB. Наиболее полную информацию о работе с программой читатель может найти в справочной системе MATLAB [3]. Для этого надо в окне справки MATLAB (окно help) открыть раздел Filter Design Toolbox Using Filter Design Toolbox Quantization Tool Overview.

Литература

1. Андреев И.И., Лавно А.А. MATLAB для DSP: SPTool - инструмент для расчёта цифровых фильтров и спектральной анализа сигналов // Цифровая обработка сигналов. 2000. № 2. С. 6-13.
2. Рабинер Л., Голуб Б. Теория и применение цифровой обработки сигналов. М.: МИР, 1978. 848 с.
3. Справочная система MATLAB 6. Filter Design Toolbox.