Алгоритмы анализа и обработки цифровых сигналов

Рыжков Александр,
группа 317

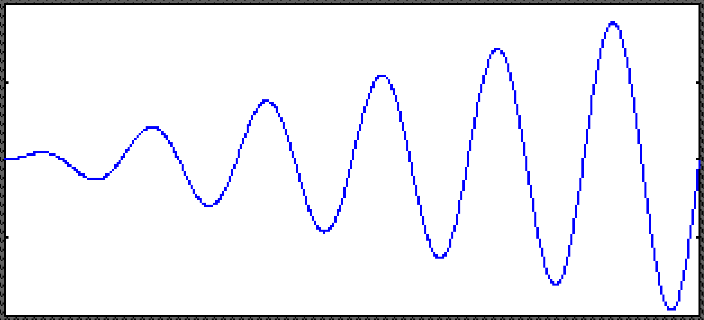
План доклада:

- Определения
- Виды сигналов
- Анализ сигнала
  - Оцифровка сигнала
  - Теорема Котельникова
  - Линейные системы
  - Преобразование Фурье
  - Спектральный анализ
  - Вейвлеты
- Обработка сигнала
  - Фильтрация
  - Шумоподавление
- Заключение

Определение сигнала:

Сигнал – некоторая функциональная зависимость от одной или нескольких переменных

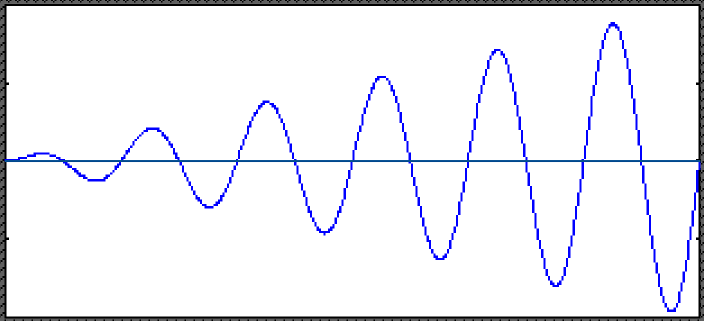
$S = F(x, y)$



$S = x(t)$

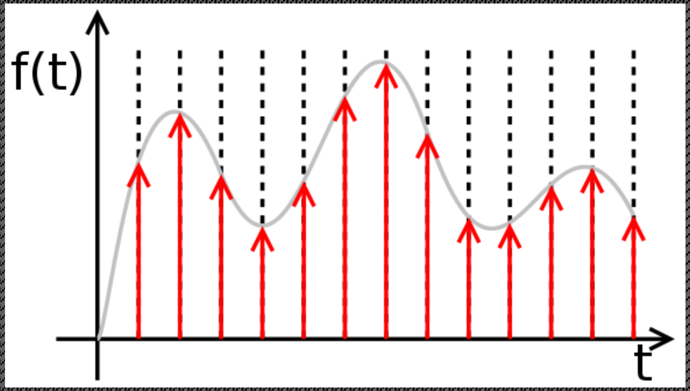
Виды сигналов:

Сигналы бывают:



Аналоговые (непрерывные)

Цифровые (дискретные)

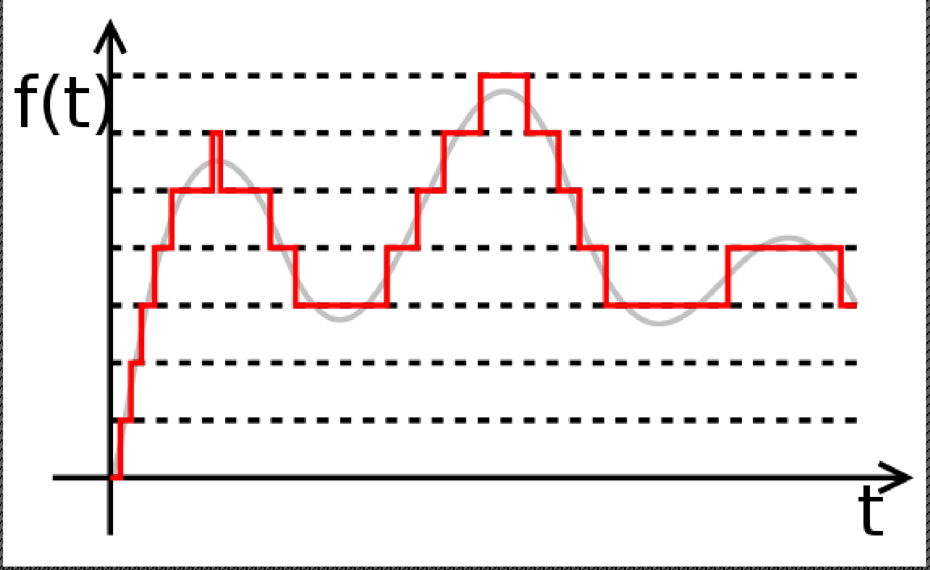


Анализ сигнала:

- Оцифровка сигнала
- Теорема Котельникова
- Линейные системы
- Преобразование Фурье
- Спектральный анализ

Дискретизация сигнала:

- Дискретизацию аналогового сигнала **S = x(t)** можно проводить двумя способами:
  - Дискретизация по частоте (по аргументу **t**)
  - Дискретизация по амплитуде (по значению функции **S**)



Дискретизация сигнала:

- Введем понятие спектра аналогового сигнала:

$$X(\nu) = \int_{-\infty}^{+\infty} x(t) \cdot e^{-2\pi i \nu t} dt \qquad x(t) = \int_{-\infty}^{+\infty} X(\nu) \cdot e^{2\pi i \nu t} d\nu$$

- Здесь x(t) – исходный сигнал, X(v) – спектр сигнала (коэффициенты при гармониках с частотой v)
- То есть мы раскладываем сигнал на синусы и косинусы с различными частотами

Теорема Котельникова:

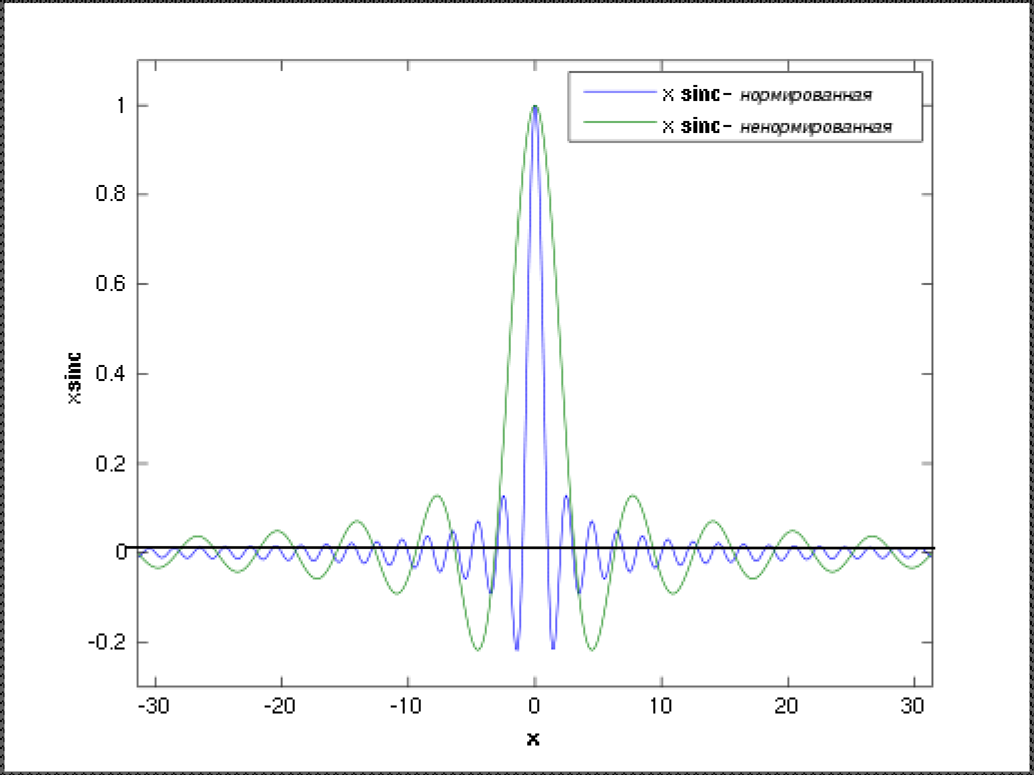
- Пусть:
  - Спектр сигнала x(t) не содержит частот выше F, т.е. X(v) = 0 за пределами отрезка [-F, F]
  - Дискретизация сигнала x(t) производится с частотой F_s, т.е. в моменты времени nT, где $T = F_s^{-1}$
  - $F_s > 2F$
- Тогда исходный аналоговый сигнал x(t) можно точно восстановить из его цифровых отсчетов x(nT), пользуясь интерполяционной формулой

$$x(t) = \sum_{n=-\infty}^{+\infty} x(nT) \cdot Sinc(t - nT)$$

$$Sinc(t) = \frac{\sin \pi F_s t}{\pi F_s t}$$

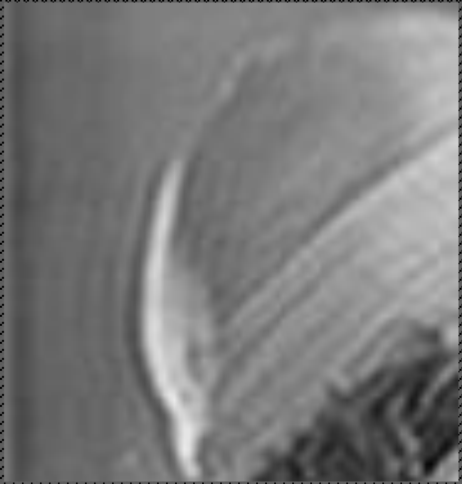
Теорема Котельникова:

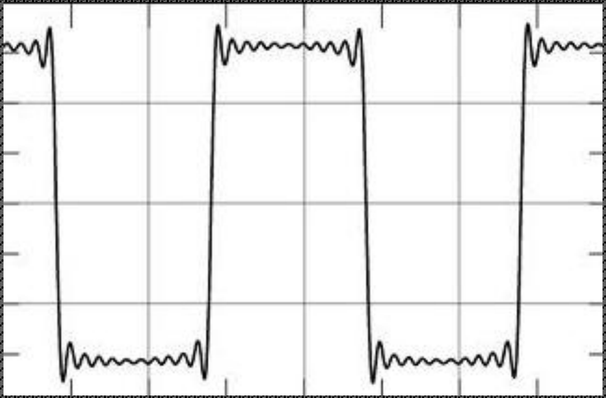
$$Sinc(t) = \frac{\sin \pi F_s t}{\pi F_s t}$$



Эффект Гиббса:

- Восстановление сигнала по такой интерполяционной формуле достаточно удобно, если бы не одно **НО**: иногда может возникать эффект Гиббса
- На изображениях проявляется в виде ореолов возле резких перепадов интенсивности



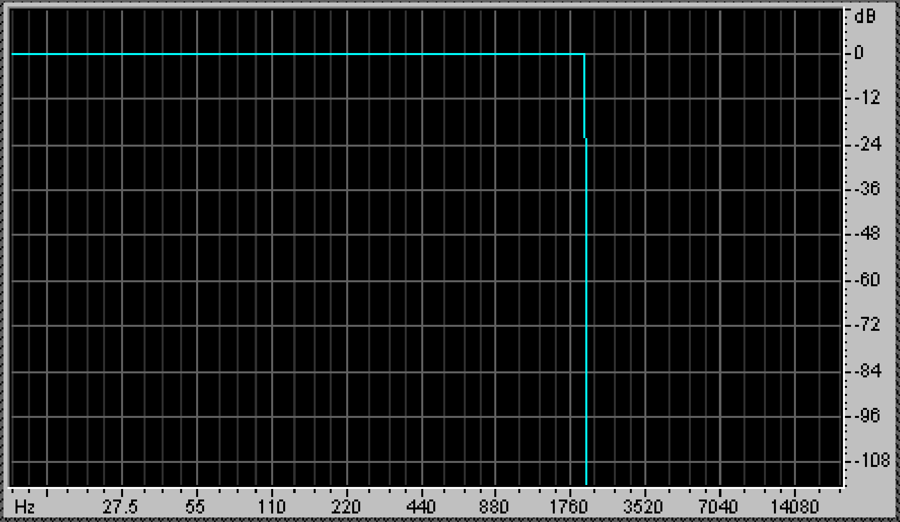


Aliasing (наложение спектра):

- Что будет с сигналом после преобразований, если условия теоремы Котельникова НЕ выполнены?
- Предположим, что сигнал не содержит частот выше 20кГц. Тогда, основываясь на теореме Котельникова, мы можем взять частоту дискретизации в 40кГц
- Пусть теперь в исходном звуке появилась помеха на 30кГц => Условия теоремы Котельникова перестали выполняться.
- Проведем дискретизацию сигнала с частотой 40кГц, а потом восстановим сигнал при помощи sinc-интерполяции
- Помеха на 30кГц отразится от половины частоты (20кГц) и попадет в слышимый диапазон на частоту в 10кГц
- Произошло наложение спектра и мы испортили исходный сигнал…

Aliasing (наложение спектра):

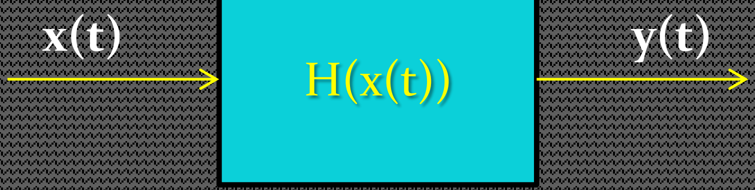
- Чтобы предотвратить Aliasing необходимо перед началом оцифровки сигнала подавить в нем все частоты ниже половины частоты дискретизации специальным анти-Aliasing фильтром



Линейные системы:

- Линейная система H(t) – некоторый преобразователь сигнала, обладающий следующими свойствами:
  - Линейность
  - Инвариантность к сдвигу

$$y(t) = H(x(t))$$



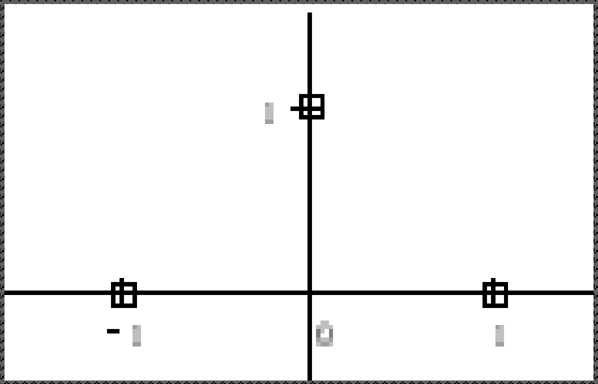
$$H(\alpha \cdot x(t)) = \alpha \cdot H(x(t))$$

$$H(x(t) + z(t)) = H(x(t)) + H(z(t))$$

$$H(x(t - t_0)) = y(t - t_0)$$

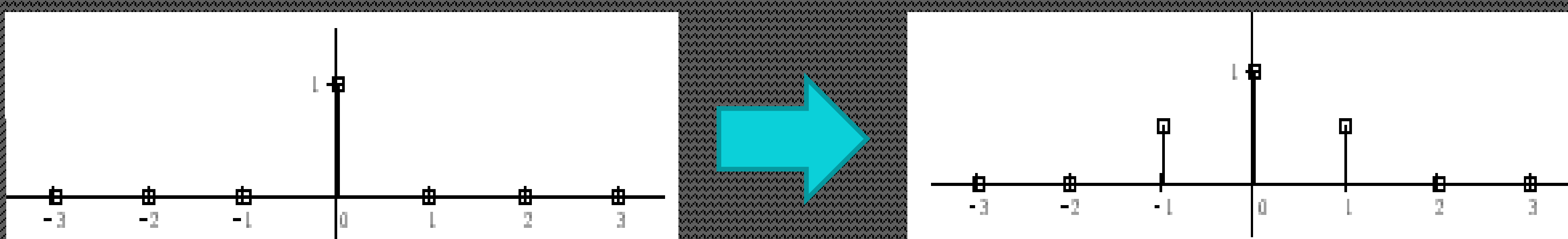
Импульсная характеристика:

- Единичный импульс δ(n)
- Разложение произвольного сигнала на сумму единичных импульсов



Импульсная характеристика:

- Отклик системы на единичный импульс δ(n) называется **импульсной характеристикой** или **импульсным откликом** системы



Выводы о линейных системах:

- Любая линейная инвариантная к сдвигу система производит операцию свертки входного сигнала со своей импульсной характеристикой.
- При подаче на любую линейную систему синусоиды, на выходе получается синусоида той же частоты, что и на входе. Измениться могут только ее амплитуда или фаза.
- Следствие: линейные системы удобно анализировать, раскладывая любые входные сигналы на синусоиды.

Преобразование Фурье:

- Преобразование Фурье – разложение исходного сигнала на различные синусоиды
- Дискретное преобразование Фурье

$$x[n] = \frac{1}{N}\sum_{k=0}^{N-1} X_k \exp\left(\frac{2\pi i k n}{N}\right)$$

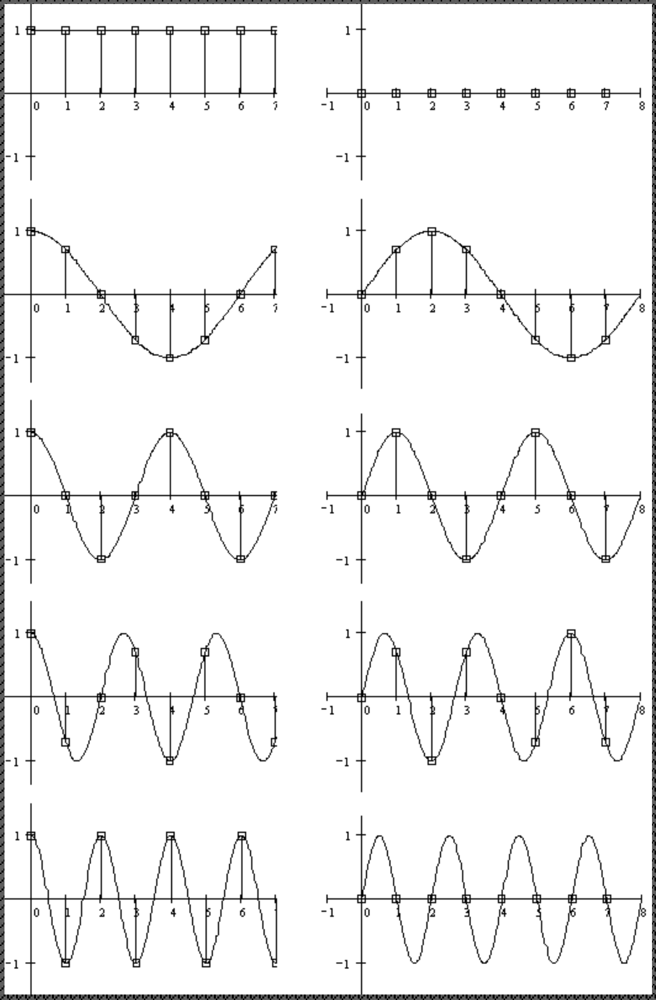
- Если исходный сигнал – вещественный, то формула записывается в следующем виде:

$$x[n] = \sum_{k=0}^{N/2} C_k \cos\frac{2\pi k(n+\varphi_k)}{N} = \sum_{k=0}^{N/2} A_k \cos\frac{2\pi kn}{N} + \sum_{k=0}^{N/2} B_k \sin\frac{2\pi kn}{N}$$

- Преобразование Фурье обратимо, то есть существуют прямое и обратное преобразования Фурье

Преобразование Фурье:

- Так выглядят базисные функции преобразования Фурье для N = 8
- У нас есть также N / 2 + 1 = 5 базисных частот
- Получим, что у нас N + 2 базисные функции, 2 из которых тождественно равны нулю
- На каждой из них у нас N чисел



Преобразование Фурье:

- Базисные функции преобразования Фурье образуют N-мерный ортогональный базис в пространстве N-мерных векторов сигналов
- Поэтому очевидно, что такое преобразование обратимо, то есть по коэффициентам A_k и B_k можно точно восстановить исходный сигнал
- Делается это обратным преобразованием Фурье – вычислением суммы конечного ряда Фурье (по сути необходимо сложить N штук N-точечных синусоид с присущими им коэффициентами)

Преобразование Фурье:

- Прямое преобразование Фурье – умножение отсчетов исходного сигнала на соответствующие синусы и косинусы:

$$A_k = \frac{2}{N}\sum_{i=0}^{N-1} x[i]\cos\frac{2\pi ki}{N} \qquad k = 1,\ldots,\frac{N}{2}-1$$

$$A_k = \frac{1}{N}\sum_{i=0}^{N-1} x[i]\cos\frac{2\pi ki}{N} \qquad k = 0,\frac{N}{2}$$

$$B_k = \frac{2}{N}\sum_{i=0}^{N-1} x[i]\sin\frac{2\pi ki}{N} \qquad k = 0,\ldots,\frac{N}{2}$$

- Для реализации этого требуется примерно N * N умножений… Медленно как то, хочется побыстрее…

Быстрое преобразование Фурье:

- Быстрое преобразование Фурье (FFT) – ускорение вычисления ДПФ
- Основной принцип - периодичность базисных функций => много одинаковых множителей, которые можно каждый раз не вычислять
- Основные преимущества:
  - Математическая точность (ошибки округления меньше, т.к. меньше число операций)
  - Число умножений порядка $N \cdot \log_2 N$, намного меньше, чем N^2 при достаточно больших N
- Ограничение – большинство реализаций этого метода принимают только вектора размерности 2^m
- Существует также и обратное быстрое преобразование Фурье (IFFT)

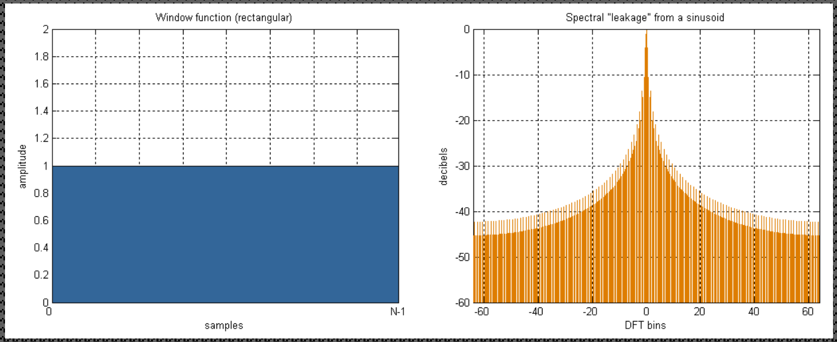
Быстрое преобразование Фурье:

- Быстрое преобразование Фурье (FFT) принимает на входе два аргумента:
  - Число $N = 2^m$
  - Вектор отсчетов исходного сигнала (который может быть записан как в вещественном, так и в комплексном варианте)
- На выходе алгоритма мы получаем:
  - Вектор коэффициентов A_k и B_k (которые могут быть записаны как в вещественном, так и в комплексном варианте ($A_k + i*B_k$))

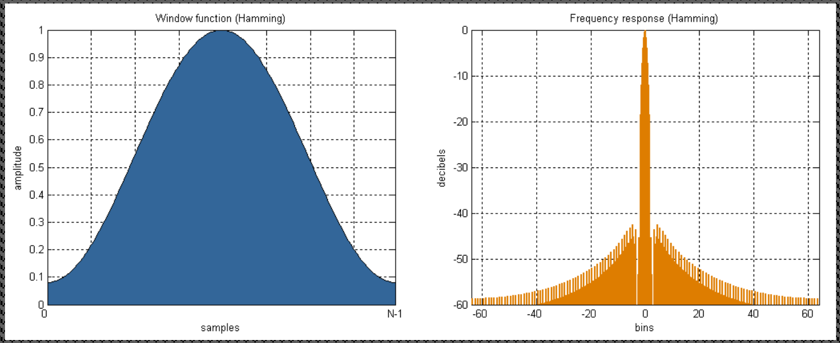
Спектральный анализ:

- Хотелось бы посмотреть на спектр сигнала, но как его отобразить?
- Применим алгоритм:
  - Возьмем нужный нам отрезок длины 2^m (отрезок меньшей длины можно дополнить нулями)
  - Если необходимо как-то улучшить изображение спектра, получаемое на выходе, то умножим сигнал на весовое окно
  - Вычислим FFT имеющегося сигнала
  - Переведем полученные коэффициенты в полярную форму, чтобы получить их амплитуды и фазы
  - Получим график зависимости амплитуды от частоты

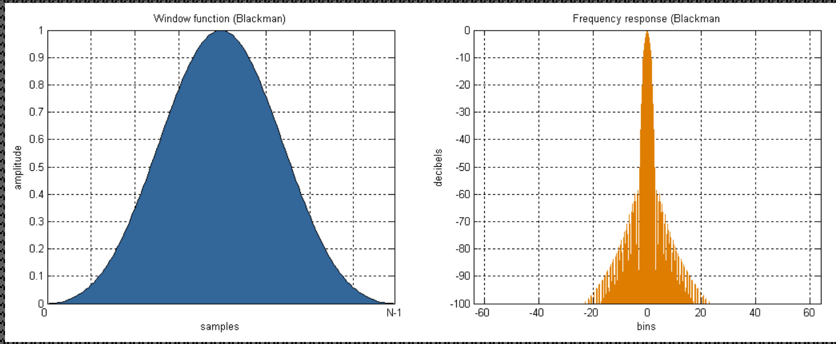
Примеры весовых окон:



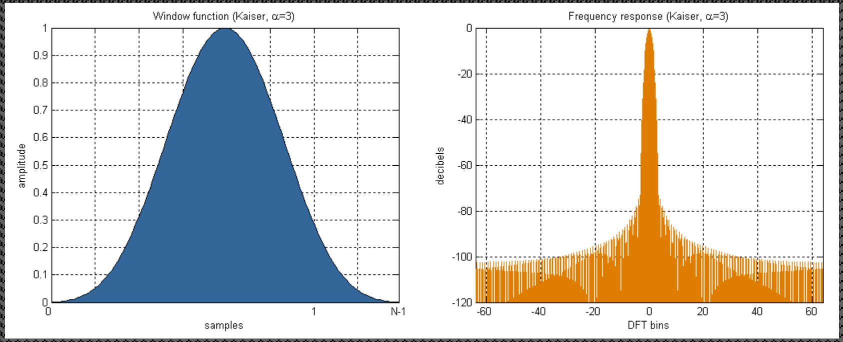
Прямоугольное (нет окна)



Hamming



Blackman

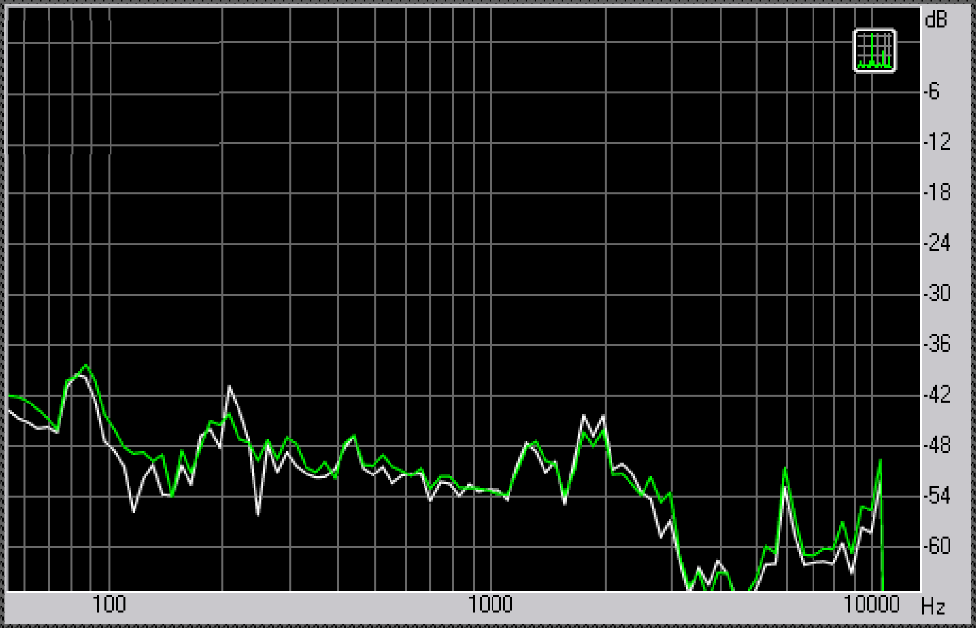


Kaiser

Спектральный анализ:

- Спектрограмма (сонограмма)
- Понятие децибела

$$A_{dB} = 10 \lg \frac{A}{A_0}$$



Оконное преобразование Фурье:

- Short Time Fourier Transform (STFT)
- Непрерывный случай:

$$F(t,\omega) = \int_{-\infty}^{\infty} f(\tau) W(\tau - t) e^{-i\omega\tau}\, d\tau,$$

- Дискретный случай:

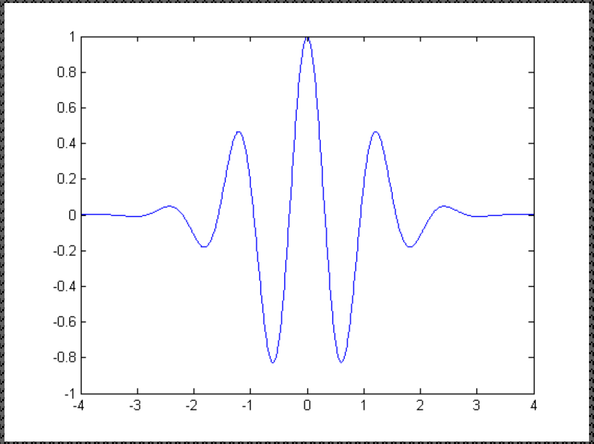
$$F(m,\omega) = \sum_{n=-\infty}^{\infty} f[n] w[n-m] e^{-j\omega n}$$

- W(τ – t) – функция весового окна

Понятие вейвлета:

- Вейвлеты – это сдвинутые и масштабированные копии $\psi_{a,b}(t)$ («дочерние вейвлеты») некоторой быстро затухающей осциллирующей функции $\psi(t)$ («материнского вейвлета»)

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}}\psi\left(\frac{t-b}{a}\right)$$



- Вейвлеты обычно используются для изучения частотного состава функций в различных масштабах и для разложения/синтеза функций в компрессии и обработке сигналов

Условия, накладываемые на ψ(t):

- Интегрируемость

$$\int_{-\infty}^{\infty} |\psi(t)| dt < \infty \qquad \int_{-\infty}^{\infty} |\psi(t)|^2 dt < \infty$$

- Нулевое среднее

$$\int_{-\infty}^{\infty} \psi(t) dt = 0$$

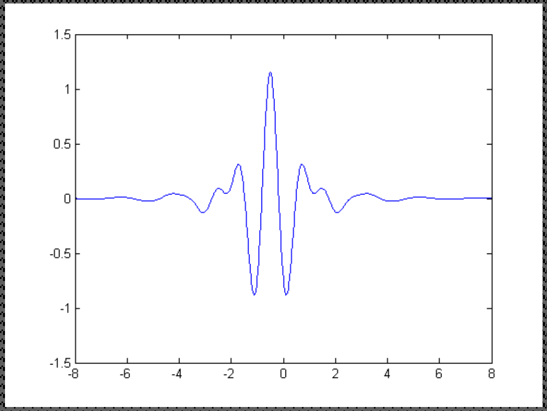
- Нормировка на всей прямой

$$\int_{-\infty}^{\infty} |\psi(t)|^2 dt = 1$$

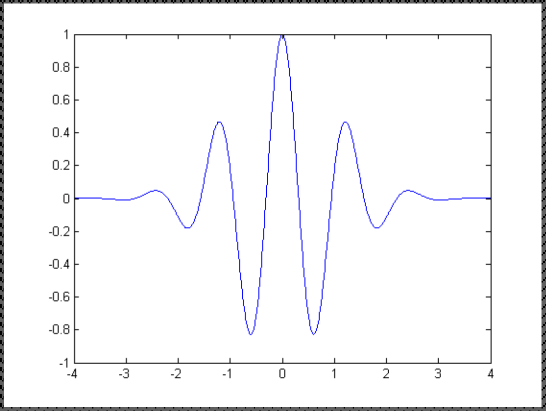
- Нулевые моменты

$$\int_{-\infty}^{\infty} t^m \psi(t) dt = 0$$

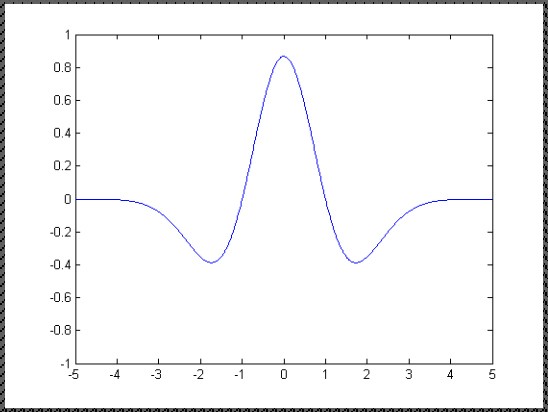
Примеры вейвлетов:



Meyer



Mortlet

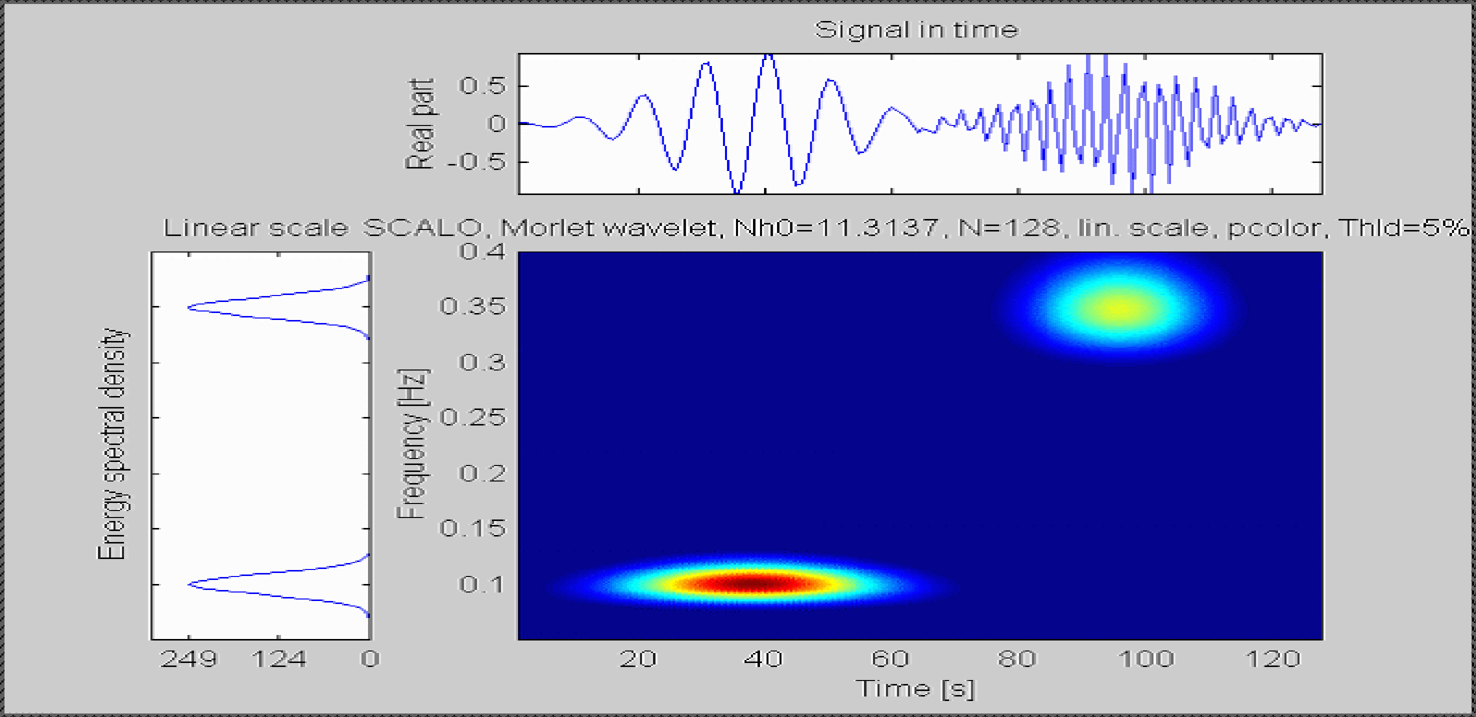


Mexican hat

Непрерывное преобразование (CWT):

- Вычисление скалярных произведений исходной функции и вейвлета

$$W_\psi\{x\}(a,b) = \left\langle x, \psi_{a,b} \right\rangle = \int_{-\infty}^{+\infty} x(t)\psi_{a,b}(t)dt$$



Дискретное преобразование (DWT):

- Дискретный вейвлет – это последовательность чисел $h_2[m]$, ортогональная своим сдвигам на четное число точек:

$$\sum_{m=-\infty}^{+\infty} h_2[m]h_2[m+2k]=0, \quad \forall k\in Z, \quad k\neq 0$$

- Также существует скейлинг-функция (НЧ-фильтр), ортогональная вейвлету:

$$\sum_{m=-\infty}^{+\infty} h_1[m]h_2[m]=0$$

Преобразование Хаара:

- Это одно из наиболее простых вейвлет-преобразований:

$$x_1^*[n] = \frac{x[n] + x[n+1]}{2} \qquad x_2^*[n] = \frac{x[n] - x[n+1]}{2}$$

- Очевидно, что исходный сигнал можно точно восстановить по двум полученным последовательностям:

$$x[n] = x_1^*[n] + x_2^*[n]$$

- Однако такое кодирование избыточно, так как мы получаем две последовательности вместо одной

Преобразование Хаара:

- Устранение избыточности:
  - Прореживаем полученные последовательности в два раза:

$$x_1[n] = x_1^*[2n] \qquad x_2[n] = x_2^*[2n]$$

- У нас все еще есть возможность точно восстановить исходный сигнал:

$$y_i[n] = \begin{cases} x_i\left[\dfrac{n}{2}\right], & n - \textit{четное} \\ 0, & n - \textit{нечетное} \end{cases} \qquad i = 1,\ 2$$

интерполяция нулями

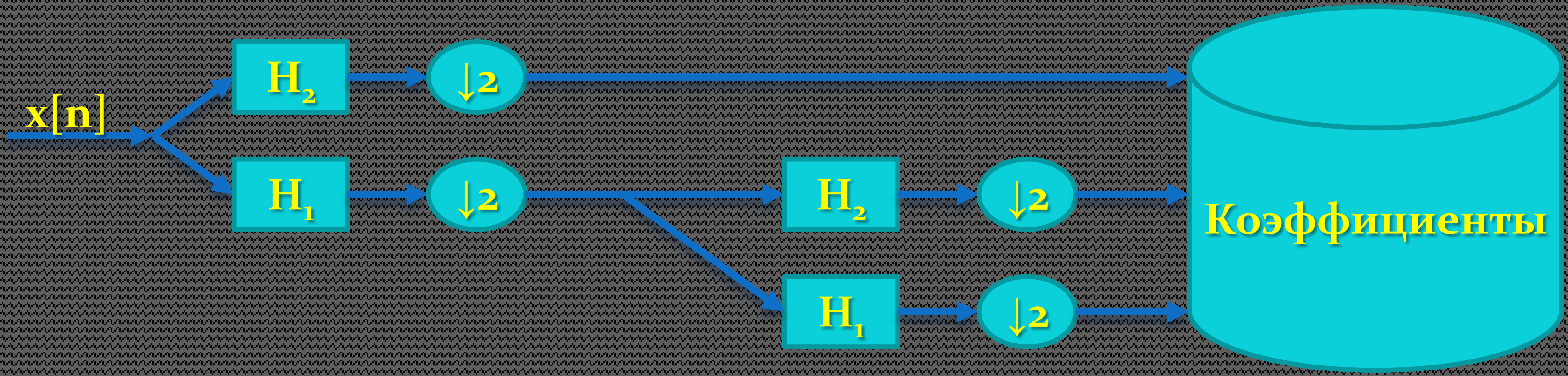
$$x_1^{**}[n] = y_1[n] + y_1[n-1] \qquad x_2^{**}[n] = y_2[n] - y_2[n-1]$$

фильтрация

$$x[n] = x_1^{**}[n] + x_2^{**}[n]$$

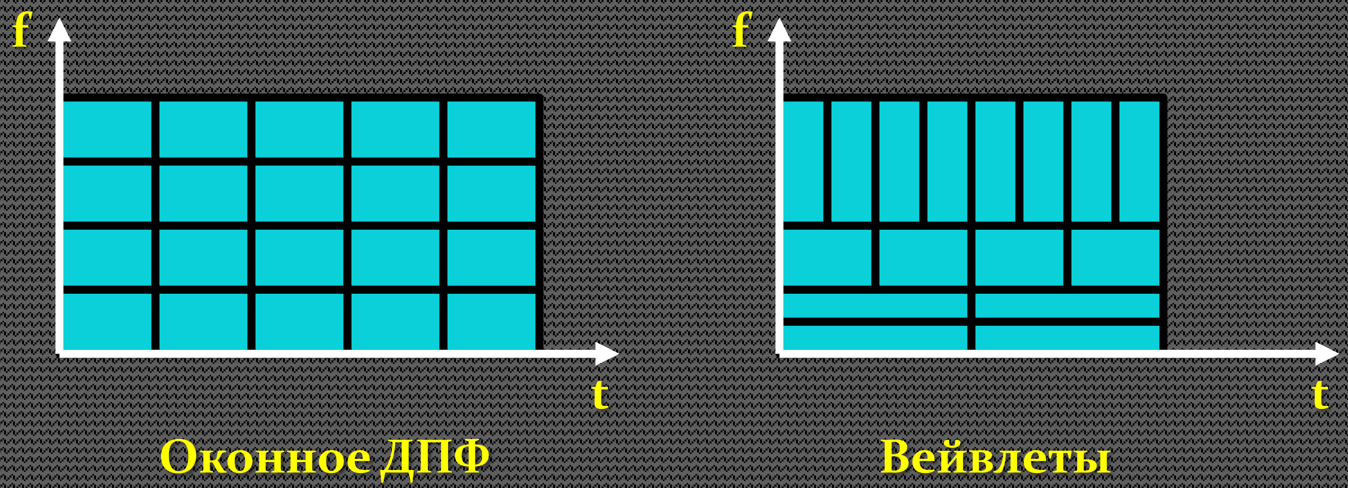
суммирование

Пирамидальное представление:



- При вейвлет-преобразовании продолжаем разложение только низкочастотных коэффициентов, так как именно низкие частоты представляют наибольший интерес и мы хотим хорошо выделить

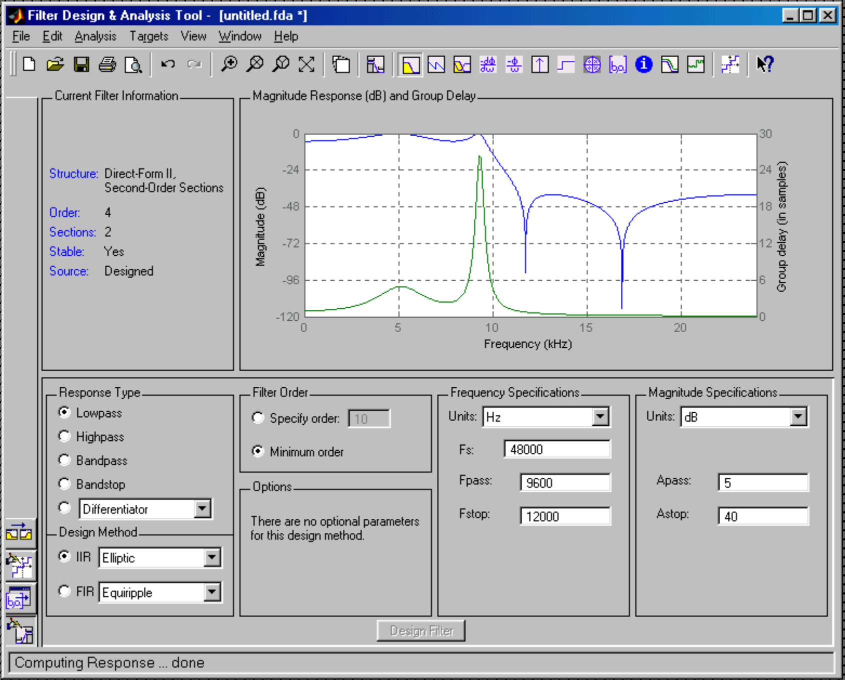
Разбиения частотно-временной области:



- Вейвлеты делят частотную ось на октавы
- Оконное ДПФ (STFT) делит частотную ось равномерно

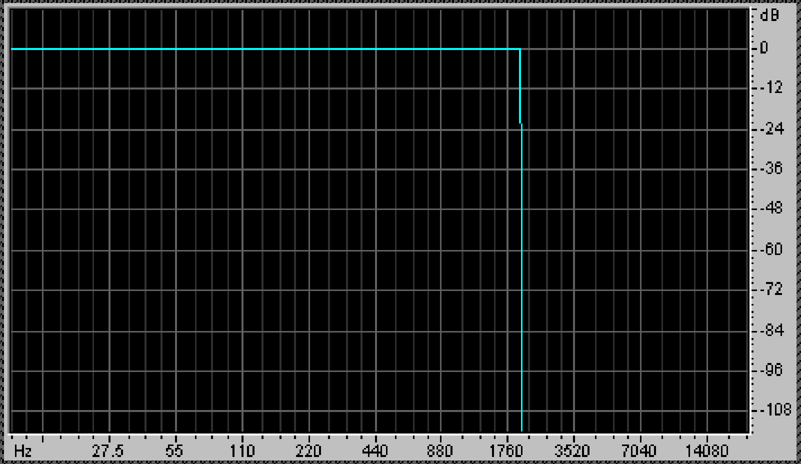
Обработка сигнала:

- Фильтрация
- Шумоподавление

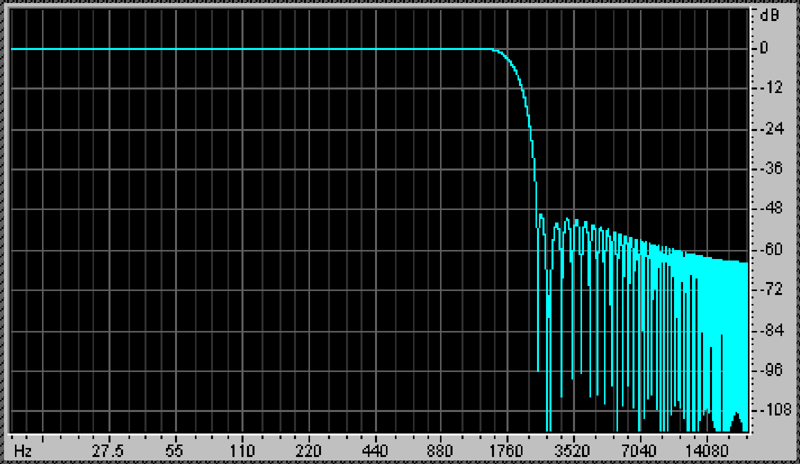


Фильтрация:

- Методы проектирования фильтров:
  - Построение фильтра с линейной фазой по произвольной заданной частотной характеристике
  - Частотная характеристика приближается к заданной с любым заданным уровнем точности
- Пример:



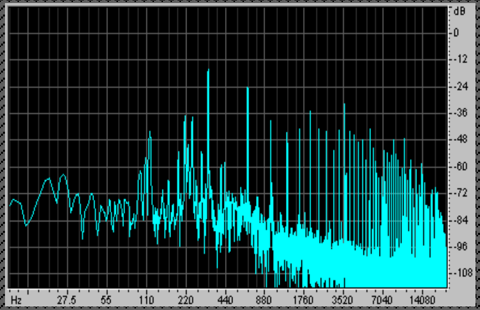
Идеальный фильтр



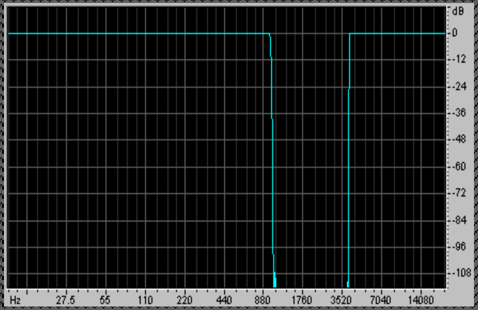
Один из реальных фильтров

Фильтрация:

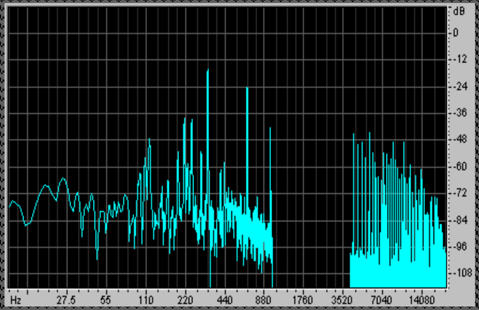
- Спектры сигналов при фильтрации (свертке) перемножаются
- Фильтры бывают идеальные (модельные) и используемые на практике
- Пример:



*



=



Шумоподавление:

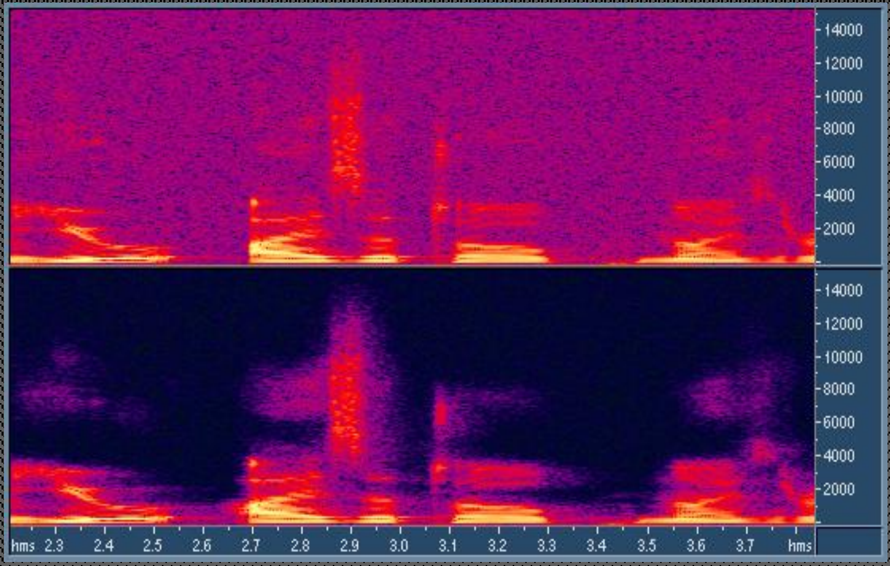
- Виды шумов:
  - Импульсные
  - Стационарные
  - Искажения
- Проблема шумоподавления по-прежнему актуальна, поскольку раньше она была из-за небольших возможностей аппаратуры, а сейчас из-за того, что многие используют некачественное, бюджетное оборудование

Шумоподавление:

- Аддитивный шум – такой шум, когда сигнал можно представить в виде:

$$dirty[n] = clean[n] + noise[n]$$

- Пример (метод спектрального вычитания):

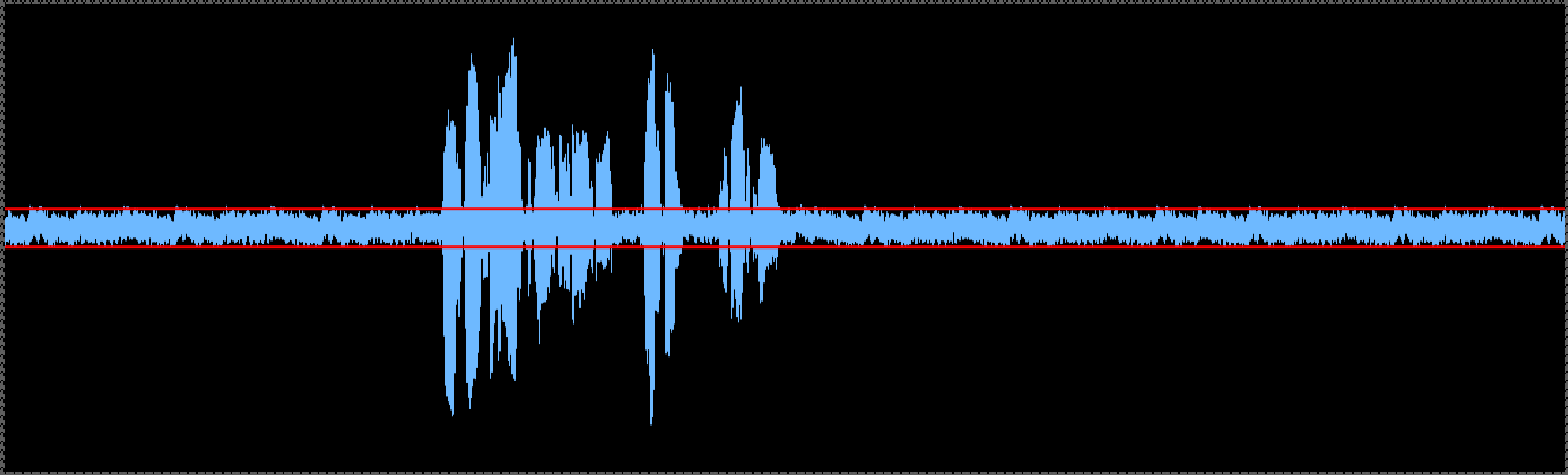


Шумоподавление:

Схема алгоритма спектрального вычитания

Шумоподавление:

- Гейт (Gate) – один из простейших методов шумоподавления – подавляет сигналы ниже определенной амплитуды
- Пример:



Заключение:

- В настоящее время анализ и обработка сигналов используется практически повсеместно
- Основные направления развития:
  - Компрессия изображений (JPEG)
  - Компрессия аудио (mp3, aac, …)
  - Мобильная телефония
  - Звукозапись
  - Шумоподавление, исправление искажений
  - Обработка и распознавание речи

Спасибо за внимание!