Володин Георгий Викторович

Быстрое преобразование Фурье (БПФ)

БПФ = быстрое ДПФ

БПФ – не является новым видом ПФ, это <u>набор алгоритмов</u> эффективного вычисления дискретного преобразования Фурье

ДПФ =>

$$\boxed{X(k) = \frac{1}{N}\sum_{n=0}^{N-1} x(n)\, e^{\frac{-j2\pi nk}{N}} = \frac{1}{N}\sum_{n=0}^{N-1} x(n)\left[\cos\frac{2\pi nk}{N} - j\sin\frac{2\pi nk}{N}\right]}$$

$$0 \le k \le N-1$$

Кол-во операций:

- N – перемножения комплексных чисел
- (N-1) – сложения комплексных чисел

Для k точки спектра!

Эффективность БПФ по сравнению с ДПФ

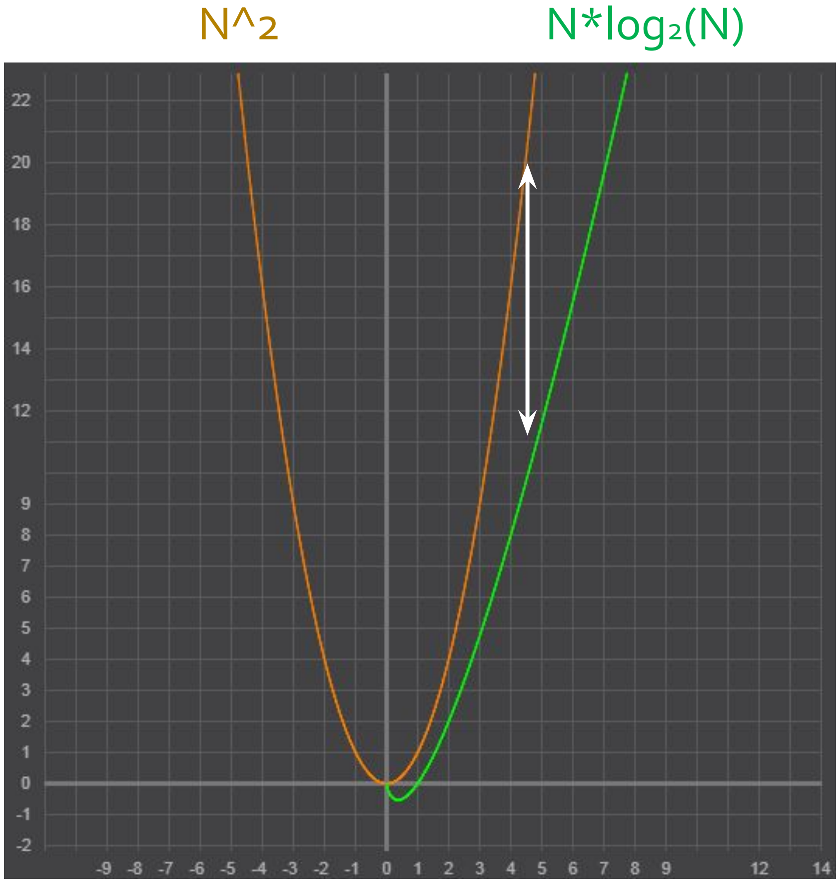
- O(N^2) – скорость вычисления для ДПФ, где мы опускаем сложение

VS

- O(N*Log2(N)) – скорость вычисления для БПФ, и скоро мы увидим почему

N	100	10^6	10^9
N^2	10000	10^12	10^18
$N*\log_2(N)$	665	2*10^7	3*10^10

Если процессор имеет скорость 1 выч/нс, то для ДПФ 10^9 будет найдено за 32 года, а с использованием ПБФ всего за 30 секунд



Проблема поворачивающего множителя в ДПФ

- $W_N = e^{-j(\frac{2\pi}{N})}$ - поворачивающий множитель

Свойства:

- Периодичность:

$$W_N^{(k+lN)(n+mN)} = W_N^{kn}$$

$$X(k) = \frac{1}{N}\sum_{n=0}^{N-1} x(n)\, e^{\frac{-j2\pi nk}{N}} = \frac{1}{N}\sum_{n=0}^{N-1} x(n)\, W_N^{nk}$$

$$W_N = e^{\frac{-j2\pi}{N}}$$

$X(0) =$	$x(0)W_8^0$	$+ x(1)W_8^0$	$+ x(2)W_8^0$	$+ x(3)W_8^0$	$+ x(4)W_8^0$	$+ x(5)W_8^0$	$+ x(6)W_8^0$	$+ x(7)W_8^0$
$X(1) =$	$x(0)W_8^0$	$+ x(1)W_8^1$	$+ x(2)W_8^2$	$+ x(3)W_8^3$	$+ x(4)W_8^4$	$+ x(5)W_8^5$	$+ x(6)W_8^6$	$+ x(7)W_8^7$
$X(2) =$	$x(0)W_8^0$	$+ x(1)W_8^2$	$+ x(2)W_8^4$	$+ x(3)W_8^6$	$+ x(4)W_8^8$	$+ x(5)W_8^{10}$	$+ x(6)W_8^{12}$	$+ x(7)W_8^{14}$
$X(3) =$	$x(0)W_8^0$	$+ x(1)W_8^3$	$+ x(2)W_8^6$	$+ x(3)W_8^9$	$+ x(4)W_8^{12}$	$+ x(5)W_8^{15}$	$+ x(6)W_8^{18}$	$+ x(7)W_8^{21}$
$X(4) =$	$x(0)W_8^0$	$+ x(1)W_8^4$	$+ x(2)W_8^8$	$+ x(3)W_8^{12}$	$+ x(4)W_8^{16}$	$+ x(5)W_8^{20}$	$+ x(6)W_8^{24}$	$+ x(7)W_8^{28}$
$X(5) =$	$x(0)W_8^0$	$+ x(1)W_8^5$	$+ x(2)W_8^{10}$	$+ x(3)W_8^{15}$	$+ x(4)W_8^{20}$	$+ x(5)W_8^{25}$	$+ x(6)W_8^{30}$	$+ x(7)W_8^{35}$
$X(6) =$	$x(0)W_8^0$	$+ x(1)W_8^6$	$+ x(2)W_8^{12}$	$+ x(3)W_8^{18}$	$+ x(4)W_8^{24}$	$+ x(5)W_8^{30}$	$+ x(6)W_8^{36}$	$+ x(7)W_8^{42}$
$X(7) =$	$x(0)W_8^0$	$+ x(1)W_8^7$	$+ x(2)W_8^{14}$	$+ x(3)W_8^{21}$	$+ x(4)W_8^{28}$	$+ x(5)W_8^{35}$	$+ x(6)W_8^{42}$	$+ x(7)W_8^{49}$

N^2 умножений с комплексными числами

- Симметрия:

$$W_N^k = -W_N^{k-N/2}$$

А давайте придумаем алгоритм вычленения повторяющихся поворачивающих множителей и будем использовать их свойства для упрощения вычислений!

(основная идея БПФ)

БПФ с прореживанием по времени

$X[k] = \sum_{k=0}^{N-1} x[n] W_N^{kn} = \sum_{n\ even} x[n] W_1^{kr} + \sum_{n\ odd} x[n] W_1^{kr}$

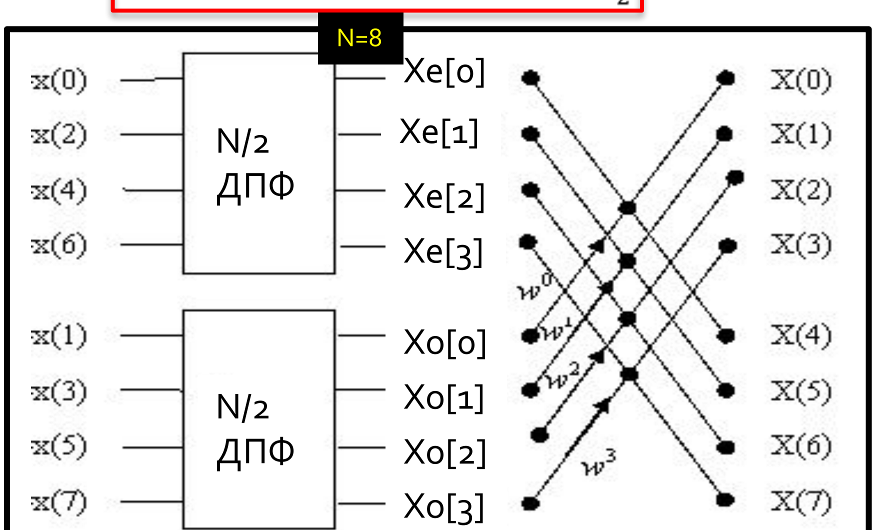
Even => n=2r , r = 0, 1, 2, …, $\frac{N}{2}$ - 1

Odd => n=2r+1

$$X[k] = \sum_{r=0}^{\frac{N}{2}-1} x[2r] W_N^{k2r} + \sum_{r=0}^{\frac{N}{2}-1} x[2r+1] W_N^{k(2r+1)} = \sum_{r=0}^{\frac{N}{2}-1} x[2r] (W_N^2)^{kr} + W_N^k \sum_{r=0}^{\frac{N}{2}-1} x[2r+1] (W_N^2)^{kr} \Rightarrow$$

$$\boxed{W_N^2 = e^{-\frac{j2\pi 2}{N}} = e^{-\frac{j2\pi}{\left(\frac{N}{2}\right)}} = W_{\frac{N}{2}}}$$

$$\Rightarrow X[k] = \underbrace{\sum_{r=0}^{\frac{N}{2}-1} x[2r] W_{\frac{N}{2}}^{kr}}_{\text{N/2 ДПФ отсчётов}} + W_N^k \underbrace{\sum_{r=0}^{\frac{N}{2}-1} x[2r+1] W_{\frac{N}{2}}^{kr}}_{\text{N/2 ДПФ отсчётов}} \Rightarrow$$



O(L)=O(2(N/2)^2+N)=O(N^2/2+N)

$$\Rightarrow X[k] = X_e[k] + W_N^k X_o[k]$$

Доопределим формулы $X_e[k]$ и $X_o[k]$ до N значений k.

$$X\left[k + \frac{N}{2}\right] = X_e\left[k + \frac{N}{2}\right] + W_N^{k+\frac{N}{2}} X_o\left[k + \frac{N}{2}\right] =$$

$$= X_e[k] - W_N^k X_o[k]$$

Построение алгоритма

$$X[k] = \begin{cases} X_e[k] + W_n^k\, X_o[k]\,, 0 \le k \le \frac{N}{2} - 1 \\ X_e\left[k - \frac{N}{2}\right] - W_n^k\, X_o\left[k - \frac{N}{2}\right], \frac{N}{2} \le k \le N - 1 \end{cases} \Rightarrow$$

Итоговая формула
для N точечного
преобразования

- Деление подобным образом в итоге уменьшило затраты вычислительной мощности. Так почему бы не продолжить этот процесс?

$$N \Rightarrow \frac{N}{2} \Rightarrow \frac{N}{4} \Rightarrow \frac{N}{8} \Rightarrow \ldots \Rightarrow \frac{N}{2^{p-1}} \Rightarrow \frac{N}{2^p} \Rightarrow 1$$

$$\boxed{\begin{gathered} \mathrm{p} = \log_2 N \\ \mathrm{N} = 2^p \end{gathered}}$$

- $1{:}\frac{N}{2} \Rightarrow 2\left(\frac{N}{2}\right)^2 + N = \frac{N^2}{2} + N$
- $2{:}\frac{N}{4} \Rightarrow 2\left(2\left(\frac{N}{4}\right)^2 + \frac{N}{2}\right) + N = \frac{N^2}{4} + 2N$
- $3{:}\frac{N}{8} \Rightarrow 2\left(2\left(2\left(\frac{N}{8}\right)^2 + \frac{N}{4}\right) + \frac{N}{2}\right) + N = \frac{N^2}{8} + 3N$

Мало при большом N

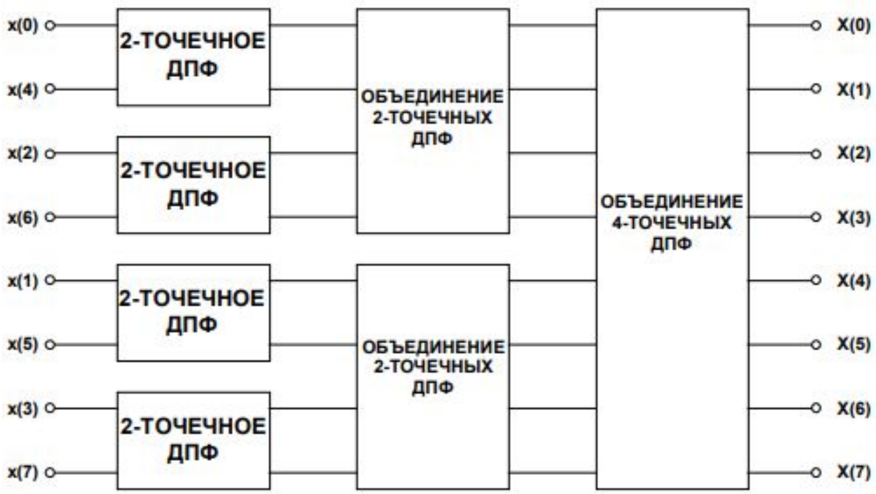
$$P{:}\frac{N}{2^p} = 1 \Rightarrow \frac{N^2}{2^p} + pN = \frac{N^2}{N} + \mathrm{N}\log_2 N$$

O(L)=O($N \log_2 N$)

(Учитывая умножения на 1/-1))

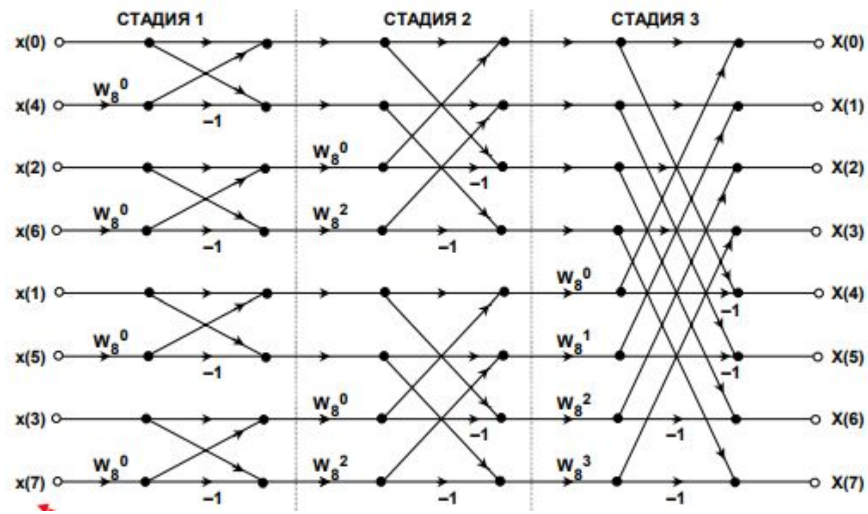
Изображение алгоритма вычисления БПФ для 8 отсчётов (прореживание по времени)

Прореживание на 4 группы 2-х точечных ДПФ



⇒

Итог



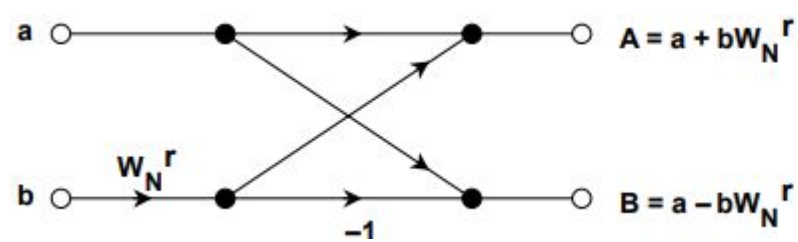
⇐

Бит-реверсное прореживание

Номер	Двоичное представление	Двоичная инверсия	Двоично-инверсный порядок
0	000	000	0
1	001	100	4
2	010	010	2
3	011	110	6
4	100	001	1
5	101	101	5
6	110	011	3
7	111	111	7

⇐

Базовая операция “бабочка” для прореживания по времени



(Несёт малую вычислительную мощность, алгоритм Рейдера)

Обратное БПФ

$$x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k) W_N^{-nk} \Rightarrow Nx^{*}(n) = \sum_{k=0}^{N-1} X^{*}(k) W_N^{nk} \Rightarrow x(n) = \frac{1}{N}\left[\sum_{k=0}^{N-1} X^{*}(k) W_N^{nk}\right]^{*}$$

ОДПФ **(*) + умножение на N** **(*) + БПФ = ОБПФ**

Обратное!

Вариации БПФ

Вычисление таблицы значений

$$W_N^{k} = \cos[(2\pi/N)k] - j\sin[(2\pi/N)k]$$

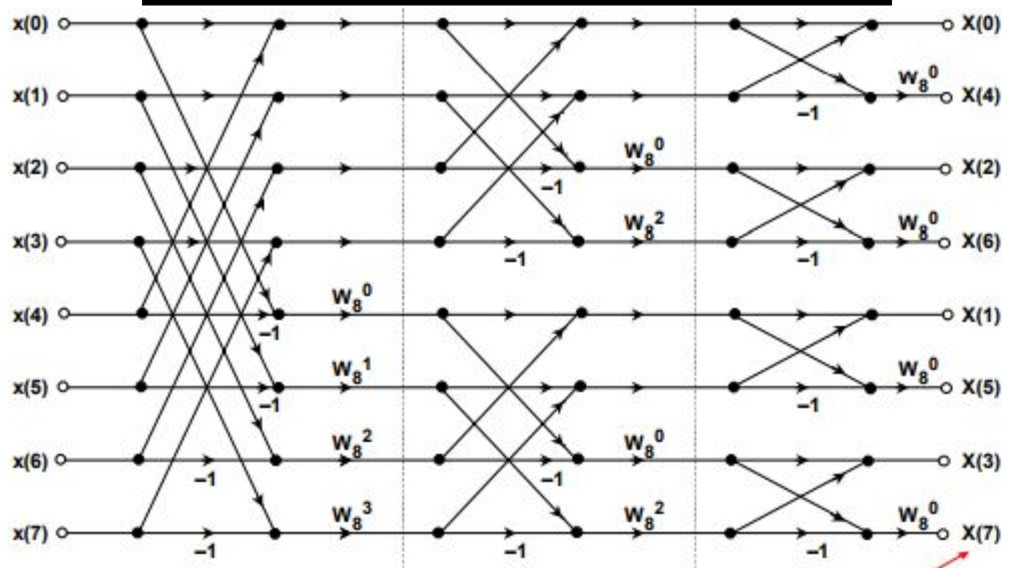
Предполагается хранение поворачивающих множителей в массиве данных.

Использование рекуррентной формулы

$$W_N^{k} = W_N^{k-l} W_N^{l}$$

Для использования этой формулы для разных этапов перемножения, достаточно предварительно вычислить и запомнить несколько пов. множ.

БПФ с прореживанием по частоте



Отсчеты спектра в бит-реверсивном порядке

$$x_1(n) = x(n), \quad n = 0,1,\ldots,N/2 - 1$$

$$x_2(n) = x(n + N/2), \quad n = 0,1,\ldots,N/2 - 1$$

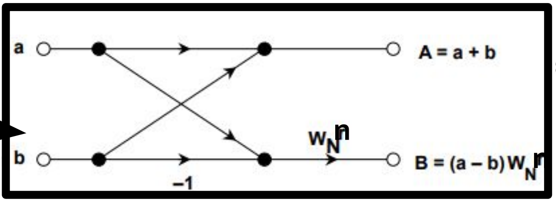
$$X(k) = \sum_{n=0}^{N/2-1} x(n)W_N^{nk} + \sum_{n=N/2}^{N-1} x(n)W_N^{nk} = \sum_{n=0}^{N/2-1} x_1(n)W_N^{nk} + \sum_{n=0}^{N/2-1} x_2(n)W_N^{(n+N/2)k} =$$

$$X(2k) = \sum_{n=0}^{N/2-1}[x_1(n) + x_2(n)](W_N^{2})^{nk} = \sum_{n=0}^{N/2-1}[x_1(n) + x_2(n)]W_{N/2}^{nk}$$

$$X(2k+1) = \sum_{n=0}^{N/2-1}[x_1(n) - x_2(n)]W_N^{n(2k+1)} = \sum_{n=0}^{N/2-1}[(x_1(n) - x_2(n))]W_N^{n}W_{N/2}^{nk}$$

ДПФ

ДПФ



$$= \sum_{n=0}^{N/2-1}[x_1(n) + e^{-j\pi k}x_2(n)]W_N^{nk}$$

Математика закончилась

Теперь можно отдохнуть на моём кресле

Минусы БПФ

- Необходимо $N = 2^m$отсчётов (можно добавить необходимое количество нулевых отсчётов – “закидывание нулями”).
- БПФ вычисляет либо все точки спектра либо ни одного! Поэтому для расчёта не полного спектра быстрее будет ДПФ.

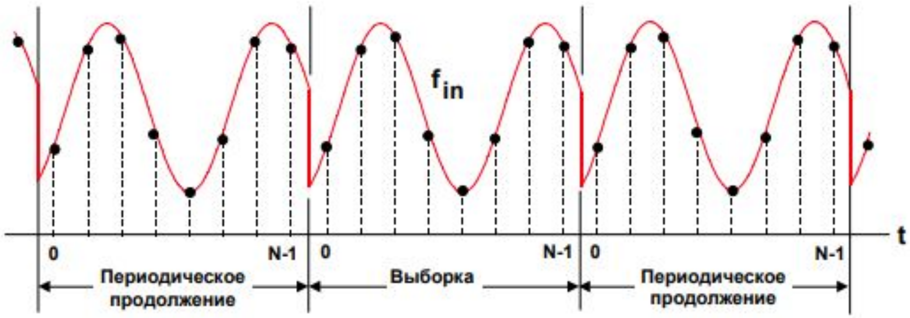
Сверх быстрое БПФ

- Это теоретически это возможно, однако это требует построение спецвычислителя (DSP – *digital signal processor*).

- Окончательная экономия времени вычисления различается для разных DSP, но алгоритм БПФ по основанию 4 может быть более чем вдвое быстрым, чем алгоритм по основанию 2 для DSP с оптимальной архитектурой.

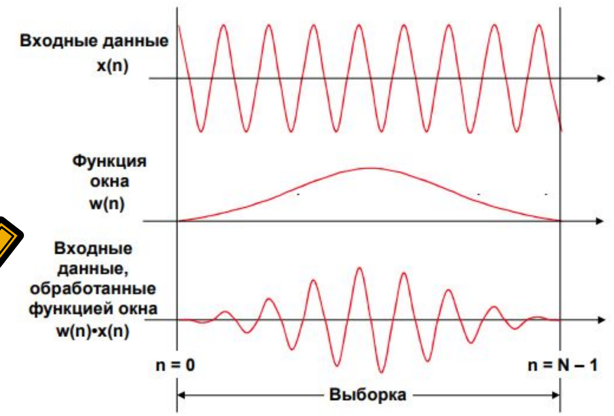
- Постоянно появляются новые концепции БПФ (УТ БПФ) которые имеют свои специфические выгоды, однако так или иначе они трудно реализуемы на сегодняшний день

БПФ и теория обработки сигналов

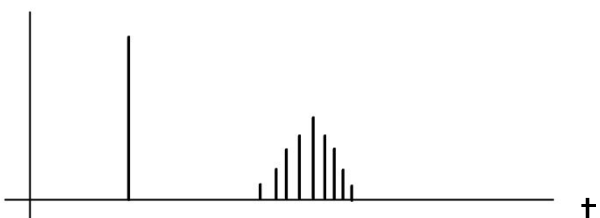


+

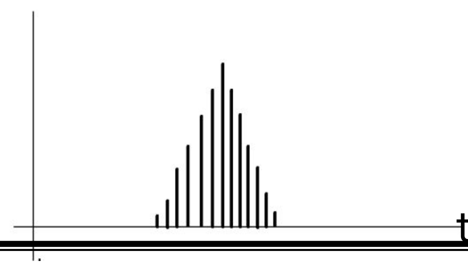
Оконное преобразование

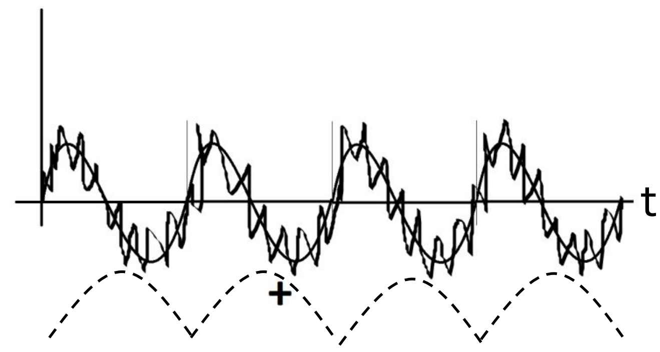


Эффект Гиббса



Эффект
“перепада громкостей”





Неопределённость:

Большая длина окна – выход из интервала гармоничности

Маленькая длина окна – искажения области низких частот

История создания БПФ

- Данное преобразование было предложено Кули и Таки (J.W.Cooley и J.W.Tukey) в 1960- ых годах и фактически являлось открытием заново идеи Рунге, Даниэльсона и Ланкоса (Runge (1903), Danielson и Lanczos (1942)).

- Джеймс Кули был нанят в IBM Thomas J. Watson Research Center в Yorktown Heights, что в Нью-Йорке. Кули работал над своим собственным проектом, когда к нему обратился Ричард Гарвин (Richard Garwin) и показал некоторые заметки Джона Тьюки (John Tukey) об алгоритме, который теоретически способен вычислять быстрое преобразование Фурье.

- Гарвин, в отличие от Кули, хорошо понимал всю важность этого алгоритма и его огромную практическую значимость, и поэтому настаивал на разработке этого алгоритма.

- Гарвин был значительно более заинтересован в улучшении дистанционного сейсмического мониторинга ядерных взрывов; русские едва ли согласились бы на проведение инспекций на их территории. Гарвин так же видел необходимость в разработке методов раннего акустического обнаружения подводных лодок.

An Algorithm for the Machine Calculation of Complex Fourier Series

By James W. Cooley and John W. Tukey

An efficient method for the calculation of the interactions of a 2^m factorial experiment was introduced by Yates and is widely known by his name. The generalization to 3^m was given by Box et al. [1]. Good [2] generalized these methods and gave elegant algorithms for which one class of applications is the calculation of Fourier series. In their full generality, Good's methods are applicable to certain problems in which one must multiply an N-vector by an $N \times N$ matrix which can be factored into m sparse matrices, where m is proportional to $\log N$. This results in a procedure requiring a number of operations proportional to $N \log N$ rather than N^2. These methods are applied here to the calculation of complex Fourier series. They are useful in situations where the number of data points is, or can be chosen to be, a highly composite number. The algorithm is here derived and presented in a rather different form. Attention is given to the choice of N. It is also shown how special advantage can be obtained in the use of a binary computer with $N = 2^m$ and how the entire calculation can be performed within the array of N data storage locations used for the given Fourier coefficients.

Список литературы

- Лекция №5 из курса “Теория Сигналов и технологии их обработки” – Уолт Кестер; Лаборатория физических основ и технологий беспроводной связи
 (http://ip-5-125.unn.ru/ftp/public/analog/5.pdf)

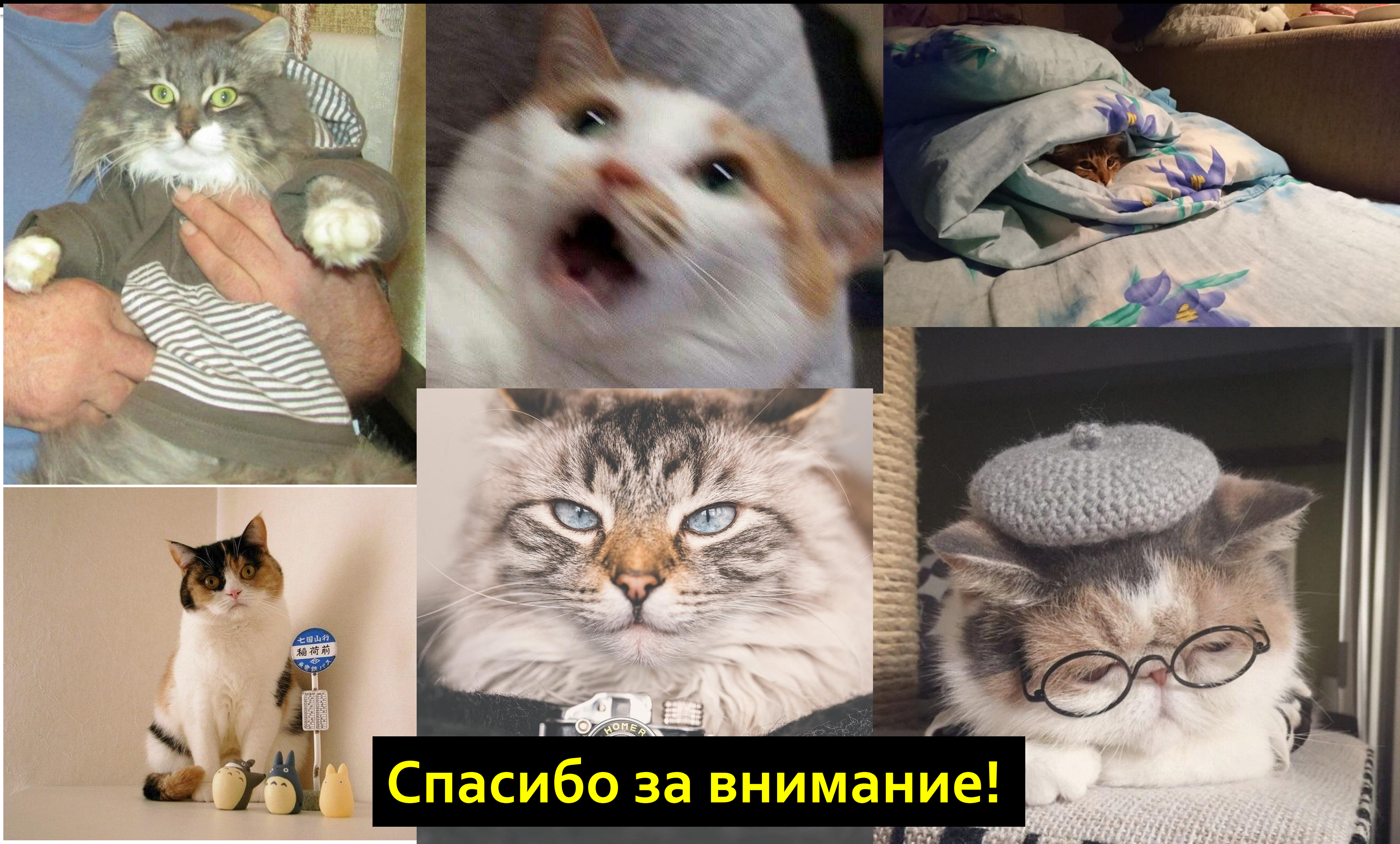
- Лекция №12 из курса “Проектирование микропроцессорных систем управления” – Денисов К.М.; Институт ЭТИПЭМС, кафедра электротехники и прецизионных электромеханических систем
(http://ets.ifmo.ru/denisov/dsp/lec12.htm)

- Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К.. Алгоритмы: построение и анализ, 2-е издание, М.:Вильямс, 2005, стр. 926-953, глава 30, «Полиномы и быстрое преобразование Фурье»

Вопросы к Зачёту

1. В чём состоит идея быстрого преобразования Фурье?
2. Преимущества и недостатки БПФ
3. Алгоритм БПФ: прореживание по времени и по частоте

Внимание!



Спасибо за внимание!