

Литературен обзор

За **Z-преобразуването**:

1. **стр. 22**, Донеvски Б., Г. Ненов, , Цифрови филтри София, Техника, 1982;

1.7. z-ПРЕОБРАЗУВАНЕ

По своята същност методът със z-преобразуването е сходен с операторния метод за изследване на аналогови, линейни и инвариантни във времето системи. Чрез него решаването на диференчните уравнения, които описват ЦЛИС, се свежда до решаване на алгебрични уравнения [5, 9, 12, 43, 104, 131, 143].

2. **стр. 85**, Боянов, Б. Цифрова обработка на сигнали – част I. Варна, БРЯГПРИНТ – АД, 2003.

тивността и определяне на коефициента на АЧХ и ФЧХ, използването на z-преобразуването за решаване на диференчни уравнения, а също така използването му за анализ на честотните свойства на ЛДИВС.

3. **стр. 238**, Lions R. G., Understanding Digital Signal Processing, PRENTICE HALL PTR, NJ, Eight Printing, 2001.

The z-transform is the discrete-time cousin of the continuous Laplace transform.[†] While the Laplace transform is used to simplify the analysis of continuous differential equations, the z-transform facilitates the analysis of discrete difference equations. Let's define the z-transform and

За **бързото преобразуване на Фурие**:

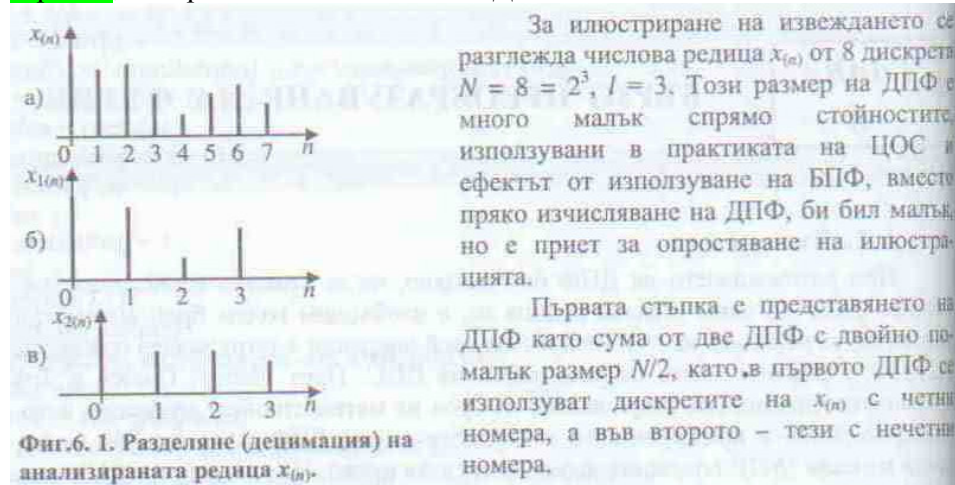
1. **стр. 35**, Боянов, Б., Е. Тодоров. Цифрова обработка на сигнали – учебно помагало. Варна, КОЛОР-ПРИНТ, 2000.

порядъка на N^2 . Значително ускоряване на изчисленията, поради намаляване на броя математически операции до $N \log_2 N$ се получава при използване на бързо преобразуване на Фурие (БПФ). Това не е друго преобразуване на Фурие, а точно е бърз алгоритъм за изчисляване на ДПФ.

2. **стр. 113**, Боянов, Б. Цифрова обработка на сигнали – част I. Варна, БРЯГПРИНТ – АД, 2003.,

номията на изчисленията. Такъв алгоритъм за изчисляване на ДПФ води до намаляване на математическите операции до число от порядъка на $N \log_2 N$ и се нарича бързо преобразуване на Фурие (БПФ). Следва да се отбележи, че БПФ не е алтернатива на ДПФ, а само алгоритъм за бързо изчисляване на ДПФ.

стр. 114, дискретен входен сигнал на ДПФ



За илюстриране на извеждането се разглежда числова редица $x(n)$ от 8 дискретни отсечки. Този размер на ДПФ е много малък спрямо стойностите, използвани в практиката на ЦОС и ефектът от използване на БПФ, вместо пряко изчисляване на ДПФ, би бил малък, но е приет за опростяване на илюстрацията.

Първата стъпка е представянето на ДПФ като сума от две ДПФ с двойно по-малък размер $N/2$, като в първото ДПФ се използват дискретите на $x(n)$ с четни номера, а във второто – тези с нечетни номера.

стр. 115, ДПФ работи с дискретен входен сигнал - $x(0)$ до $x(7)$;

$$\begin{aligned}
 X_{(0)} &= X_{1(0)} + W_8^0 \cdot X_{2(0)}, \\
 X_{(1)} &= X_{1(1)} + W_8^1 \cdot X_{2(1)}, \\
 X_{(2)} &= X_{1(2)} + W_8^2 \cdot X_{2(2)}, \\
 X_{(3)} &= X_{1(3)} + W_8^3 \cdot X_{2(3)}, \\
 X_{(4)} &= X_{1(4)} + W_8^4 \cdot X_{2(4)} = X_{1(0)} - W_8^0 \cdot X_{2(0)}, \\
 X_{(5)} &= X_{1(5)} + W_8^5 \cdot X_{2(5)} = X_{1(1)} - W_8^1 \cdot X_{2(1)}, \\
 X_{(6)} &= X_{1(6)} + W_8^6 \cdot X_{2(6)} = X_{1(2)} - W_8^2 \cdot X_{2(2)}, \\
 X_{(7)} &= X_{1(7)} + W_8^7 \cdot X_{2(7)} = X_{1(3)} - W_8^3 \cdot X_{2(3)}.
 \end{aligned} \tag{6.6}$$

От изложеното се вижда, че два дискрета на 8-точковото ДПФ $X_{(k)}$ и $X_{(k+N/2)}$ се получават чрез сумиране и извеждане на една и съща двойка дискрети $X_{1(k)}$ и $X_{2(k)}$. Тези връзки могат да се обобщят с изразите

3. стр. 129, Lions R. G., Understanding Digital Signal Processing, PRENTICE HALL PTR, Eight Printing, 2001. (БПФ е ДПФ),

excessive. In 1965 a paper was published by Cooley and Tukey describing a very efficient algorithm to implement the DFT[1]. That algorithm is now known as the fast Fourier transform (FFT).[†] Before the advent of the FFT,

стр. 54, Lions (входните сигнали на ДПФ са дискрети),

$x(0) = 0.3535, \quad x(1) = 0.3535,$
 $x(2) = 0.6464, \quad x(3) = 1.0607,$
 $x(4) = 0.3535, \quad x(5) = -1.0607,$
 $x(6) = -1.3535, \quad x(7) = -0.3535.$

139, Lions, структурна схема на БПФ-входните сигнали са цифрови, а не аналогови;

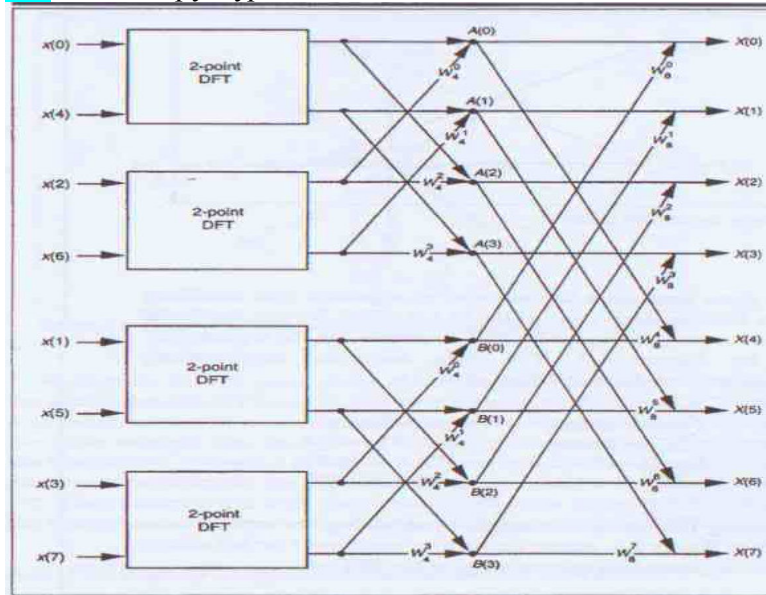


Figure 4-3 FFT implementation of an 8-point DFT as two 4-point DFTs and four 2-point DFTs.

За диференчните и диференциалните уравнения:

1. **стр. 7**, Боянов, Б., Е. Тодоров. Цифрова обработка на сигнали – учебно помагало. Варна, КОЛОР-ПРИНТ, 2000.

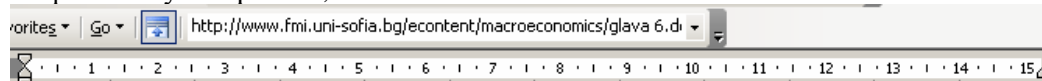
средства. В упражненията №№ 1, 2 и 3 се разглеждат основни дефиниции и характеристики на линейни дискретни инвариантни във времето системи, които са ключови за ЦОС. Разглеждат се диференчни уравнения, структурни схеми, импулсна характеристика, преобразуване на Фурие за дискретни сигнали и системи, коефициент на предаване, честотна характеристика, Z-преобразуване, предавателна функция, полюсно-нулева диаграма, устойчивост и др. С цел

2. **стр. 22**, Цифрови филтри, Доневски Б., Г. Ненов, София, Техника, 1982;

1.7. z-ПРЕОБРАЗУВАНЕ

По своята същност методът със z-преобразуването е сходен с операторния метод за изследване на аналогови, линейни и инвариантни във времето системи. Чрез него решаването на диференчните уравнения, които описват ЦЛИС, се свежда до решаване на алгебрични уравнения [5, 9, 12, 43, 104, 131, 143].

3. www.fmi.uni-sofia.bg, сайт на Факултета по математика и информатика на Софийския университет;



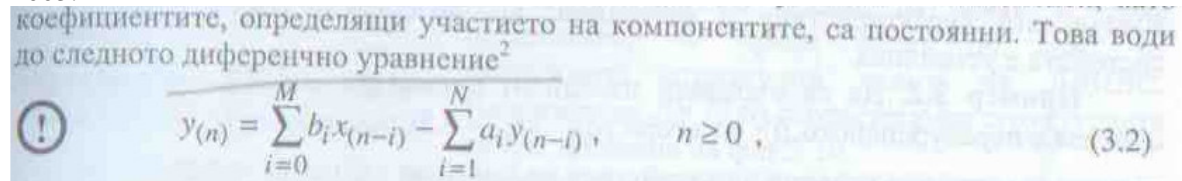
Основните математически средства, използвани при тях са **диференциални уравнения** (при непрекъснато отчитане на времето) или **диференчни уравнения** (при дискретно отчитане на времето). Голяма

4. След като в Google се напише „differential equation” се вижда как изглеждат диференциалните уравнения. Те нямат нищо общо с уравненията от ръководството:

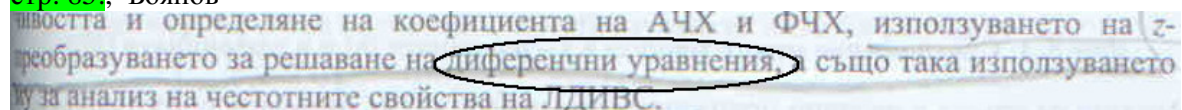
A **linear differential equation** of order n is a differential equation written in the following form:

$$a_n(x) \frac{d^n y}{dx^n} + a_{n-1}(x) \frac{d^{n-1} y}{dx^{n-1}} + \dots + a_1(x) \frac{dy}{dx} + a_0(x)y = f(x),$$

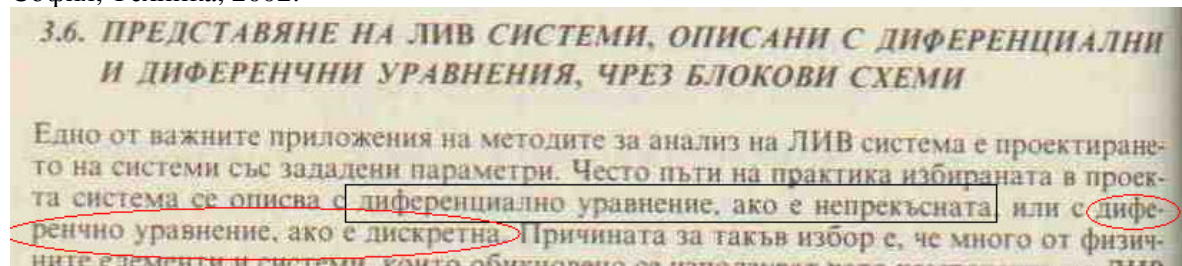
5. **стр. 25**, Боянов, Б. Цифрова обработка на сигнали – част I. Варна, БРЯГПРИНТ – АД, 2003.



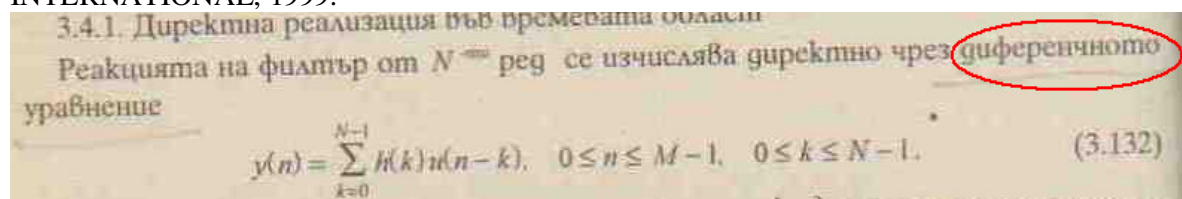
стр. 85, Боянов



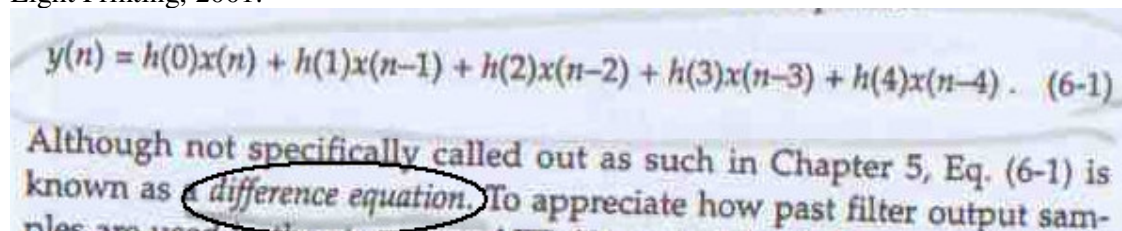
6. а ето какво пишат по въпроса на **стр. 102** Опенхайм, А. и колектив. Сигнали и системи. София, Техника, 2002.



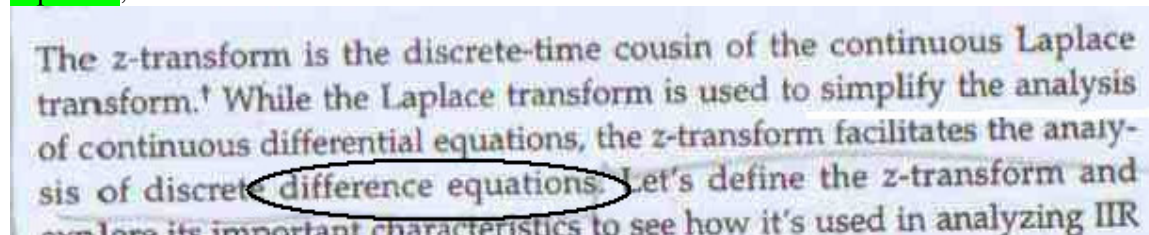
7. **стр. 76**, Иванов, Р. Цифрова обработка на едномерни сигнали. Габрово, ALMAMATER INTERNATIONAL, 1999.



8. стр. 221, Lions R. G., Understanding Digital Signal Processing, PRENTICE HALL PTR, Eight Printing, 2001.



стр. 238, Lions



Псевдокод

(http://www.proz.com/kudoz/english_to_bulgarian/computers:_software/3274987-pseudocode.html)

Definition from STING software engineering glossary (Copyright [CERN](#), European Laboratory for Particle Physics) :

A notation resembling a programming language but not intended for actual compilation. **It usually combines some of the structure of a programming language with an informal natural-language description of the computations to be carried out.** It is often produced by CASE systems as a basis for later hand coding.

Definition from own experience or research:

Условен начин на описание на алгоритми, разбираем за човека. Алгоритъм, описан по този начин, трябва да се опише на език за програмиране, за да може да се изпълни от компютър.

За справка:

Описанието на алгоритъм изисква средство за изразяване на последователността от стъпки, които трябва да се изпълнят. Най-общите средства за описание са: естествен език, псевдокод и формализиран език за програмиране. Естественият език е най-лек за ползване, но и най-неточен, тъй като не е строго формализиран и допуска двусмислие. Езиците за програмиране, като Pascal и C са точни, но са по-трудни за писане и разбиране. **Псевдокодът е полезен, защото е по средата. Той има формализирани управляващи структури, подобно на език за програмиране, но допуска описание на отделни стъпки със средствата на естествен език. Точно тези свойства на псевдокода позволяват постъпковото разработване на един алгоритъм**

<http://www.technologybulgaria.com/алгоритмите-в-информатиката.html>

Псевдокодът не е написан на истински език за програмиране. Той се състои от прости изречения, които описват какво ще прави програмата.

http://vb.teacher.bg/unit03/frame03_04.htm

Псевдокод е вид програмен код, който е написан на измислен/неконкретен/несъществуващ език с цел да се очертае структурата на даден алгоритъм или въобще решение - т.е. някаква скица на програма, без да се задълбава в детайли около език, реализация на техническите детайли и т.н.

Пример за псевдокод:

Алгоритъм: пиене на бира

1. Купувате бирата
2. Намирате отварачка и я отваряте
3. Пиете.