



# ΕΠΕΞΕΡΓΑΣΙΑ ΣΗΜΑΤΩΝ

## DFT - FFT ΔΙΑΚΡΙΤΟΣ ΚΑΙ ΤΑΧΥΣ ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΣ FOURIER

ΚΑΘΗΓΗΤΗΣ ΑΘΑΝΑΣΙΟΣ ΣΚΟΔΡΑΣ

skodras@upatras.gr  
www.ece.upatras.gr/skodras

ΠΕΜΠΤΗ .11.2023 - ΩΡΑ 9:00-11:00

### DFT

#### ΔΙΑΚΡΙΤΟΣ ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΣ FOURIER (DFT - DISCRETE FOURIER TRANSFORM)

ΟΡΙΣΜΟΙ

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j \frac{2\pi}{N} kn}, \quad k=0,1,\dots,N-1$$
$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j \frac{2\pi}{N} kn}, \quad n=0,1,\dots,N-1$$

↔ ΕΘΩΣ μετασχ. (DFT)

↔ ΑΝΤΙΣΤΡΑΦΟΣ μετασχ. (IDFT)

- Οι ακολουθίες  $x(n)$  και  $X(k)$  είναι περιοδικές με περίοδο  $N$  (βλ. υποσημείωση)
- Οι συντελεστές  $X(k)$  αποτελούν ουσιαστικά δείγματα του  $X(e^{j\omega})$  σε ισαπέχοντα κατά  $\frac{2\pi}{N}$  δείγματα στον άξονα των συχνοτήτων, δηλαδή  $X(k)$  είναι η συντομογραφία του  $X(e^{j\frac{2\pi}{N}k})$ .
- Για τον υπολογισμό κάθε σφηκίου του DFT, ή αλλιώς λόγω για τον υπολογισμό κάθε συντελεστή  $X(k)$ , απαιτούνται  $N$  πολλαπλασιασμοί και  $(N-1)$  πολλαπλασιασμοί πρόσθεσης. Συνεπώς, για τον υπολογισμό των  $N$  συντελεστών απαιτούνται συνολικά  $N^2$  πολλαπλασιασμοί και  $N(N-1)$  πολλαπλασιασμοί πρόσθεσης. Σημειώνεται ότι για κάθε πολλαπλασιασμό απαιτούνται 4 πολλαπλασιασμοί και 2 πρόσθεσεις πραγματικών αριθμών.

Υποσημείωση:  $x(n+N) = x(n)$  για όλα τα  $n$   
 $X(k+N) = X(k)$  για όλα τα  $k$

## DFT

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j\frac{2\pi}{N}kn}, \quad k=0,1,\dots,N-1$$

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j\frac{2\pi}{N}kn}, \quad n=0,1,\dots,N-1$$

← όπως forward. (DFT)

← αντίστροφος forward. (IDFT)

Ορίζοντας με  $W_N = e^{-j\frac{2\pi}{N}}$  την  $N$ -οστή ρίζα της μονάδας, οι παραπάνω εκφράσεις εκφράζονται ως εξής:

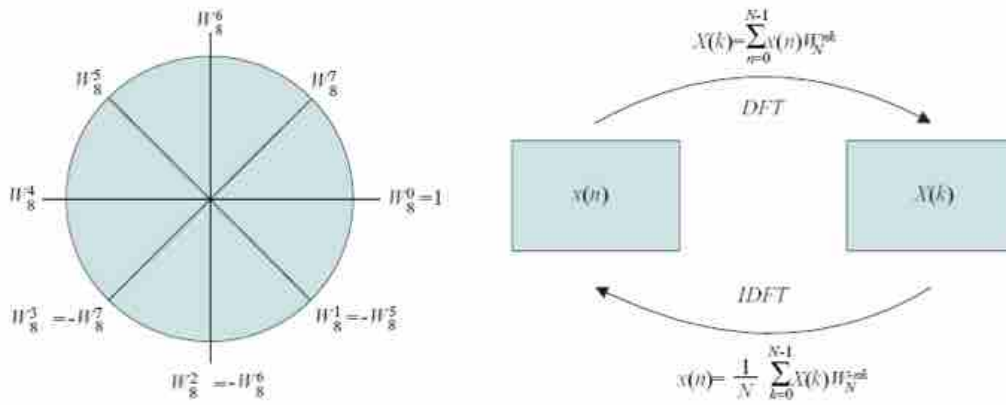
$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn}, \quad k=0,1,\dots,N-1$$

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) W_N^{-kn}, \quad n=0,1,\dots,N-1$$

← DFT

← IDFT

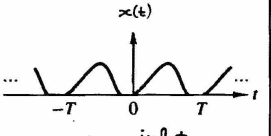
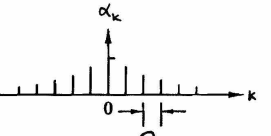
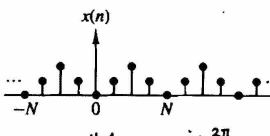
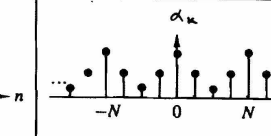
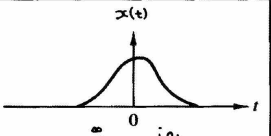
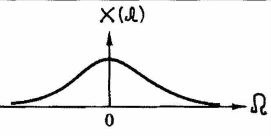
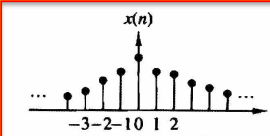
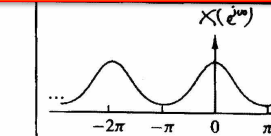
$$W_N = e^{-j2\pi/N}$$



- Οι ορίσμοι του DFT και IDFT  $N$ -συμμετρικών μπορούν να εκφραστούν και με  $n$  φορές πίνακων ως ακολούθως:

$$\begin{aligned}
 \mathbf{X}_N &= \mathbf{W}_N \mathbf{x}_N & \rightarrow & \begin{bmatrix} X(0) \\ X(1) \\ \vdots \\ X(N-1) \end{bmatrix} = \begin{bmatrix} W_N^{00} & W_N^{01} & \dots \\ W_N^{10} & W_N^{11} & \dots \\ \vdots & \vdots & \ddots \\ W_N^{(N-1)0} & \dots & \dots \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ \vdots \\ x(N-1) \end{bmatrix} \\
 \mathbf{x}_N &= \frac{1}{N} \mathbf{W}_N^* \mathbf{X}_N
 \end{aligned}$$

Ο πίνακας  $\mathbf{W}_N$  είναι συμμετρικός και ορθογώνιος (unitary).

		Continuous-time signals		Discrete-time signals	
		Time-domain	Frequency-domain	Time-domain	Frequency-domain
Periodic signals	Fourier series	 $\alpha_k = \frac{1}{T} \int_T x(t) e^{-jk\Omega_0 t} dt$	 $x(t) = \sum_{k=-\infty}^{\infty} \alpha_k e^{jk\Omega_0 t}$	 $\alpha_k = \frac{1}{N} \sum_{n=0}^{N-1} x(n) e^{-jk \frac{2\pi}{N} n}$	 $x(n) = \sum_{k=0}^{N-1} \alpha_k e^{jk \frac{2\pi}{N} n}$
		Continuous and periodic	Discrete and aperiodic	Discrete and periodic	Discrete and periodic
Aperiodic signals	Fourier transforms	 $X(\Omega) = \int_{-\infty}^{\infty} x(t) e^{-j\Omega t} dt$	 $x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\Omega) e^{j\Omega t} d\Omega$	 $X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x(n) e^{-j\omega n}$	 $x(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(e^{j\omega}) e^{j\omega n} d\omega$
		Continuous and aperiodic	Continuous and aperiodic	Discrete and aperiodic	Continuous and periodic

Σημείωση:  $\omega = \Omega T$

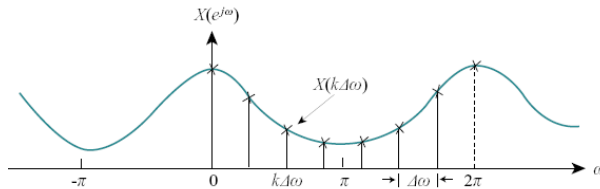
### Discrete Fourier Transform (DFT)

$$X(e^{j\omega}) = \sum_{n=0}^{N-1} x(n) e^{-j\omega n}, \quad 0 \leq \omega \leq 2\pi$$

$$X(k) \equiv X\left(\frac{2\pi k}{N}\right) = X(e^{j\omega}) \Big|_{\omega=2\pi k/N} = \sum_{n=0}^{N-1} x(n) e^{-j2\pi kn/N}, \quad k=0, 1, \dots, N-1$$

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j2\pi kn/N}, \quad n=0, 1, \dots, N-1$$

DFT pair



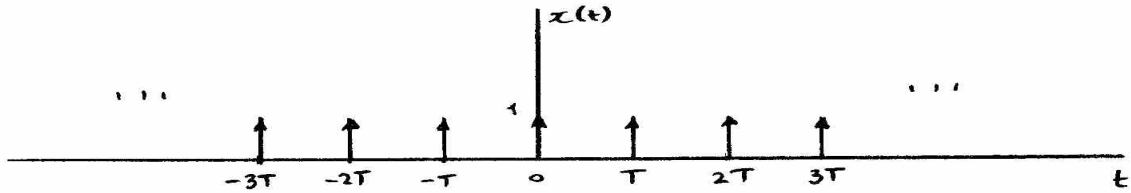
$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{nk}, \quad k=0, 1, \dots, N-1,$$

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

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

## ΦΑΣΜΑ ΤΡΑΙΝΟΥ ΚΡΟΥΣΤΙΚΩΝ

Να υπολογιστεί ο ΜΦ του περιοδικού τρένου κρουστικών.



## ΦΑΣΜΑ ΤΡΑΙΝΟΥ ΚΡΟΥΣΤΙΚΩΝ

Να υπολογιστεί ο ΜΦ του περιοδικού τρένου κρουστικών.

Λύση

το σήμα αυτό εκφράζεται ως

$$x(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT)$$

όπου  $T$  είναι η περίοδος του σήματος.

$$\omega_0 = \frac{2\pi}{T}$$

## ΦΑΣΜΑ ΤΡΑΙΝΟΥ ΚΡΟΥΣΤΙΚΩΝ

Να υπολογιστεί ο ΜΦ του περιοδικού τρένου κρουστικών.

Λύση

το σήμα αυτό εκφράζεται ως

$$x(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT)$$

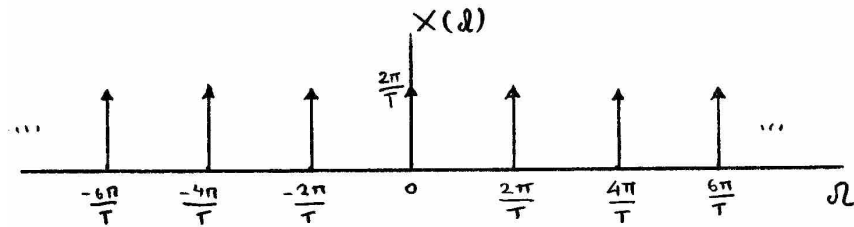
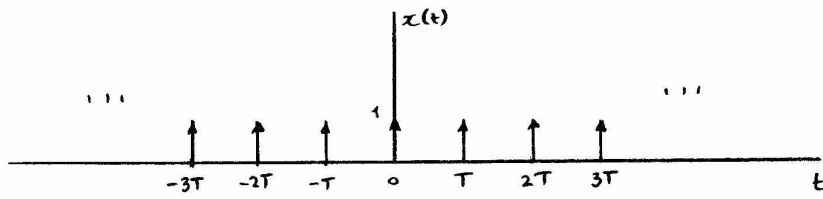
όπου  $T$  είναι η περίοδος του σήματος. Οι συντελεστές της σειράς Fourier υπολογίζονται ως  $\omega_0 = \frac{2\pi}{T}$

$$a_k = \frac{1}{T} \int_{-\infty}^{\infty} x(t) e^{-jk\omega_0 t} dt = \frac{1}{T} \int_{-T/2}^{+T/2} \delta(t) e^{-jk\omega_0 t} dt = \frac{1}{T}$$

Με άλλα λόγια, καθίνας από τους συντελεστές της σειράς Fourier του περιοδικού τρένου κρουστικών έχει πλάτος  $1/T$ . Άρα, ο ΜΦ ισούται με:

$$X(\omega) = \sum_{k=-\infty}^{\infty} 2\pi a_k \delta(\omega - k\omega_0) \Leftrightarrow X(\omega) = \frac{2\pi}{T} \sum_{k=-\infty}^{\infty} \delta(\omega - k\frac{2\pi}{T})$$

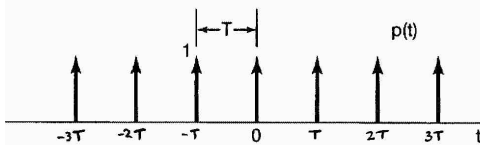
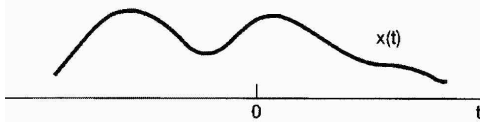
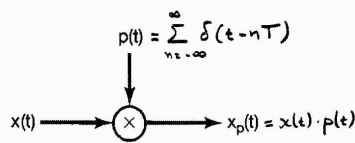
Παρατηρούμε ότι ο ΜΦ του περιοδικού τρένου κρουστικών περίοδου  $T$  ισούται με ένα περιοδικό τρένο κρουστικών στη συχνότητα, περίοδου  $\frac{2\pi}{T}$ .



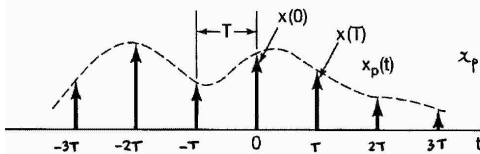
Γίνεται φανερό ότι καθώς η περίοδος  $T$  αυξάνεται, δηλαδή η απόσταση μεταξύ των κρουστικών στον χρόνο αυξάνεται, η αντίστοιχη απόσταση μεταξύ των κρουστικών στη συχνότητα μειώνεται, δηλαδή η βασική συχνότητα μειώνεται.

## Δειγματοληψία: Πεδίο του Χρόνου

### ΠΕΔΙΟ ΧΡΟΝΟΥ

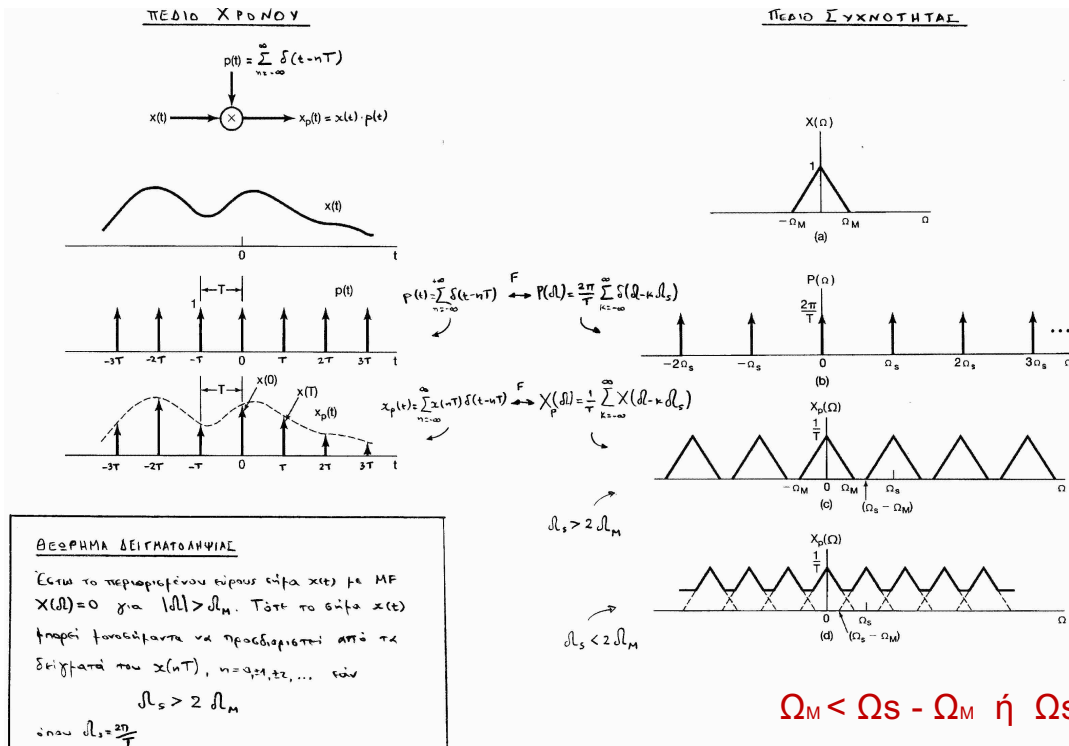
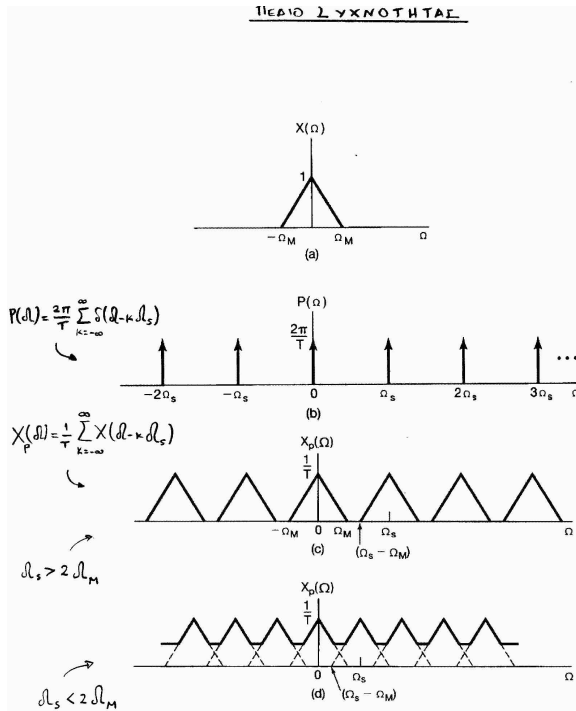


$$p(t) = \sum_{n=-\infty}^{\infty} \delta(t-nT)$$



$$x_p(t) = \sum_{n=-\infty}^{\infty} x(nT) \delta(t-nT)$$

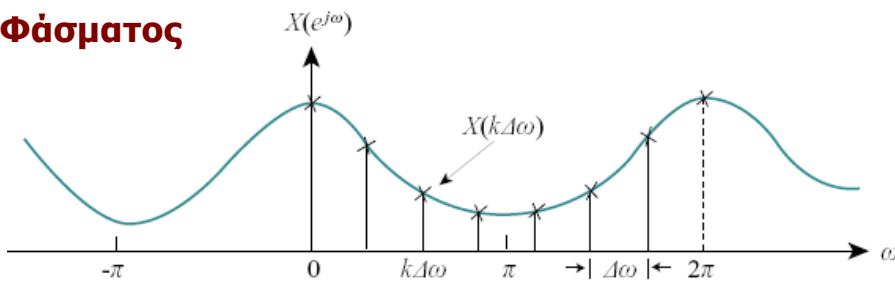
# Δειγματοληψία: Πεδίο της Συχνότητας





## Δειγματοληψία

### Φάσματος

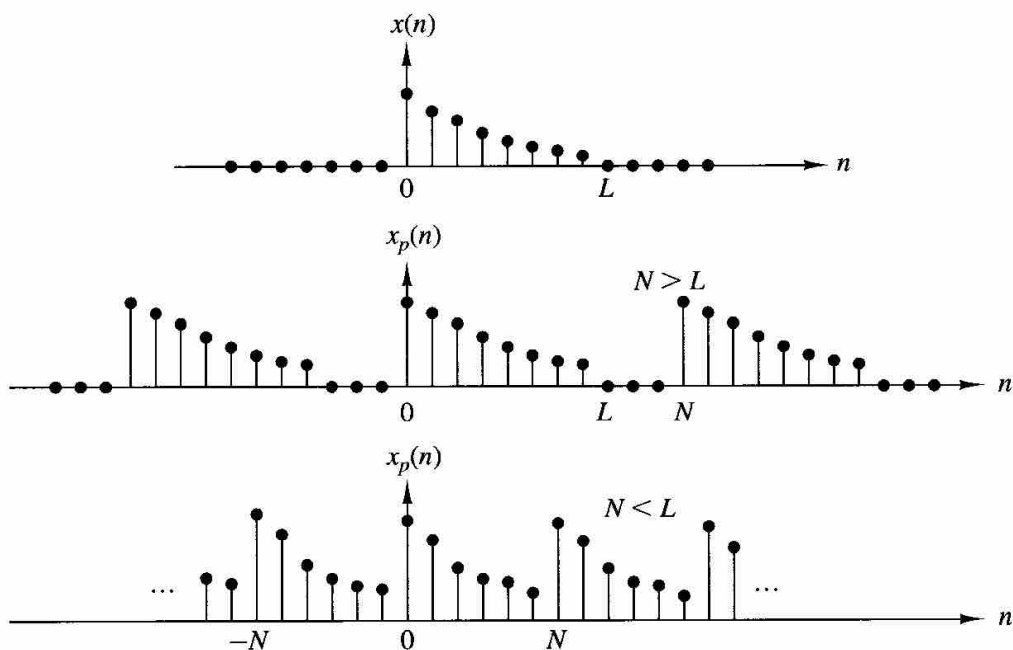


Η δειγματοληψία ενός περιοδικού φάσματος αντιστοιχεί σε ένα περιοδικό σήμα διακριτού χρόνου.

Στην ουσία πρόκειται για τη **συνέλιξη** του σήματος  $x(n)$  με μια σειρά από συναρτήσεις δέλτα με περίοδο  $N$ , δηλαδή μια πράξη που δημιουργεί αντίγραφα του  $x(n)$  ανά  $N$  δείγματα.

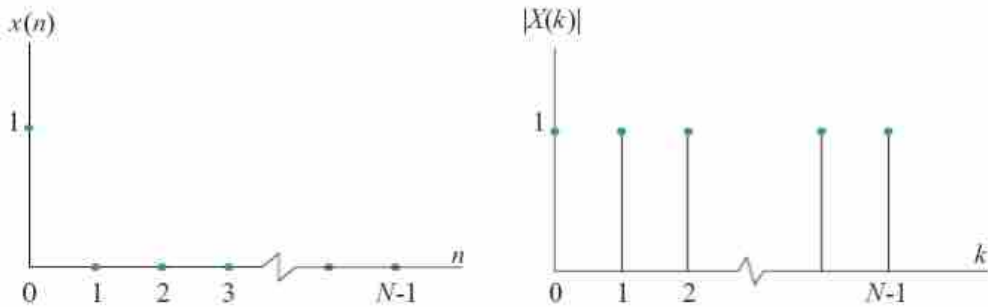
Το αperiοδικό σήμα  $x(n)$  είναι **πεπερασμένης διάρκειας  $L$** . Για την αποφυγή του φαινομένου **time-aliasing** πρέπει  $N \geq L$ .

### Παράδειγμα Time-Aliasing



Example:  $x(n) = \delta(n)$

$$X(k) = \sum_{n=0}^{N-1} x(n)W_N^{nk} = \sum_{n=0}^{N-1} \delta(n)W_N^{nk} = \delta(0)W_N^{0k} + \delta(1)W_N^{1k} + \dots + \delta(N-1)W_N^{(N-1)k} = 1 \cdot 1 + 0 + 0 + \dots + 0 = 1$$



Example  $x(n) = \begin{cases} A, & n = 0, 1, \dots, N-1 \\ 0, & \text{άλλοι} \end{cases}$

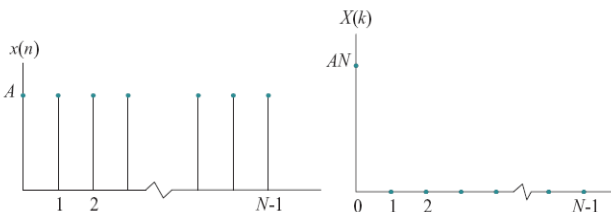
$$X(k) = \sum_{n=0}^{N-1} x(n)W_N^{nk} = \sum_{n=0}^{N-1} A e^{-j\frac{2\pi}{N}nk}, k = 0, 1, 2, 3, \dots, N-1$$

Για  $k=0$  προκύπτει:  $X(0) = \sum_{n=0}^{N-1} A e^{-j\frac{2\pi}{N}n \cdot 0} = \sum_{n=0}^{N-1} A \cdot 1 = A \sum_{n=0}^{N-1} 1 = A \cdot N$

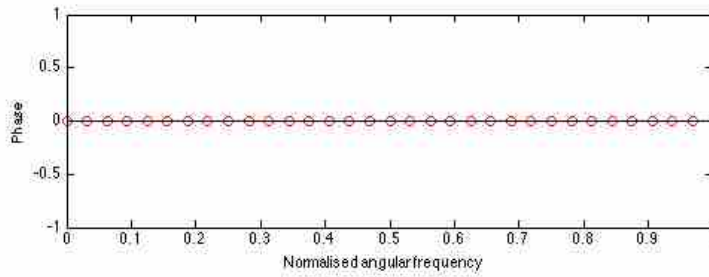
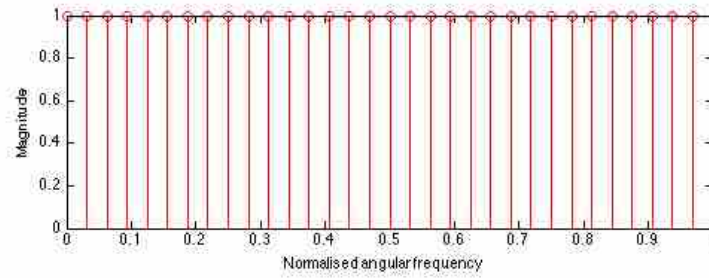
Για  $k \neq 0$  προκύπτει:  $X(k) = \sum_{n=0}^{N-1} A e^{-j\frac{2\pi}{N}nk} = A \sum_{n=0}^{N-1} e^{-j\frac{2\pi}{N}nk}$ ,  $k = 1, 2, 3, \dots, N-1$

$$X(k) = A \cdot \frac{1 - e^{-j\left(\frac{2\pi}{N}k\right)N}}{1 - e^{-j\left(\frac{2\pi}{N}k\right)}} = A \cdot \frac{1 - e^{-j2\pi k}}{1 - e^{-j\frac{2\pi k}{N}}} = 0 \text{ για } k = 1, 2, 3, \dots, N-1$$

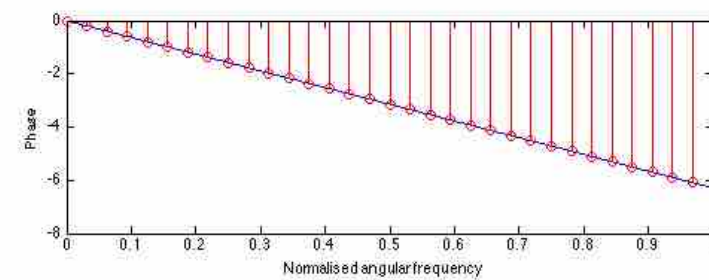
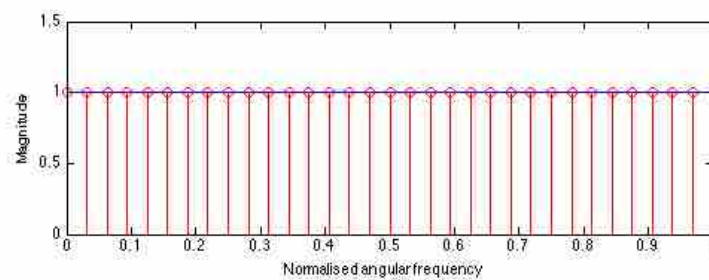
$$X(k) = AN\delta(k), k = 0, 1, 2, \dots, N-1$$



### Παράδειγμα DFT (N=32): $x(n)=\delta(n)$



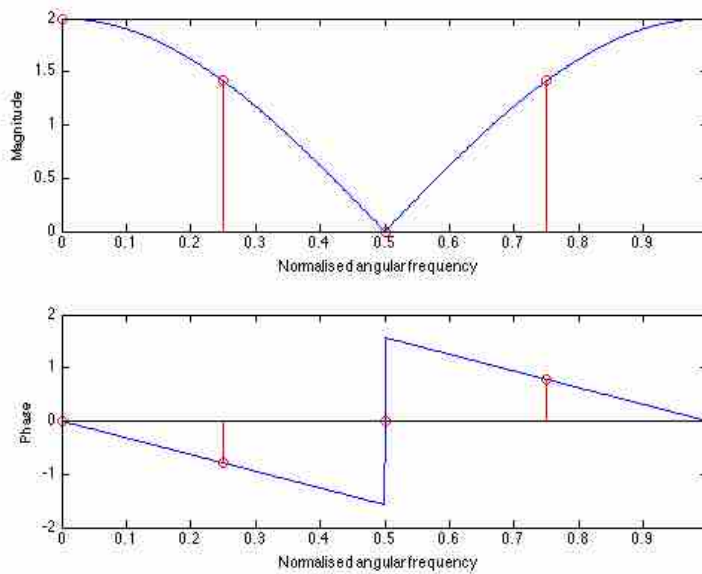
### Παράδειγμα DFT (N=32): $x(n)=\delta(n-1)$



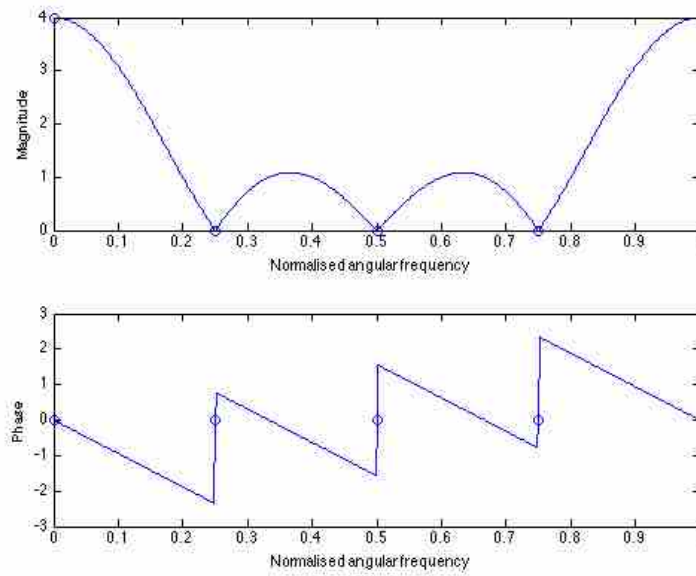
# ΙΔΙΟΤΗΤΕΣ DFT

Ιδιότητα	Ακολουθία μήκους N	DFT N σημείων
	$x_1(n)$	$X_1(k)$
	$x_2(n)$	$X_2(k)$
Γραμμικότητα	$a_1x_1(n) + a_2x_2(n)$	$a_1X_1(k) + a_2X_2(k)$
Κυκλική μετατόπιση στο χρόνο	$x(n - n_0)$	$W^{kn} X(k)$
Κυκλική μετατόπιση στη συχνότητα	$W^{-kn} x(n)$	$X(k - k_0)$
Συζυγής ακολουθία	$x^*(n)$	$X^*(-k)$
Κατοπτρισμός στο χρόνο	$x(-n)$	$X^*(k)$
Κυκλική συνέλιξη	$x_1(n) \otimes x_2(n)$ <sup>[1]</sup>	$X_1(k)X_2(k)$
Πολλαπλασιασμός	$x_1(n)x_2(n)$	$\frac{1}{N} \sum_{m=0}^{N-1} X_1(m)X_2(k-m)$
Συζυγής συμμετρία για πραγματικά σήματα	$x(n)$ πραγματική	$\begin{cases} \bar{X}(k) = X^*(-k) \\ \text{Re}\{X(k)\} = \text{Re}\{X(-k)\} \\ \text{Im}\{X(k)\} = -\text{Im}\{X(-k)\} \\  X(k)  =  X(-k)  \\ \angle X(k) = -\angle X(-k) \end{cases}$
Θεώρημα του Parseval	$\sum_{n=0}^{N-1}  x(n) ^2 = \frac{1}{N} \sum_{k=0}^{N-1}  X(k) ^2$	

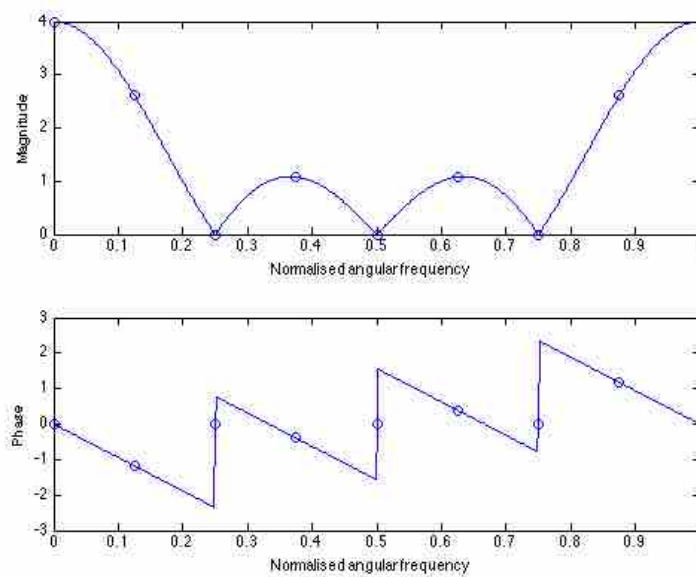
## Παράδειγμα DFT (N=4): $x(n) = \delta(n) + \delta(n-1)$



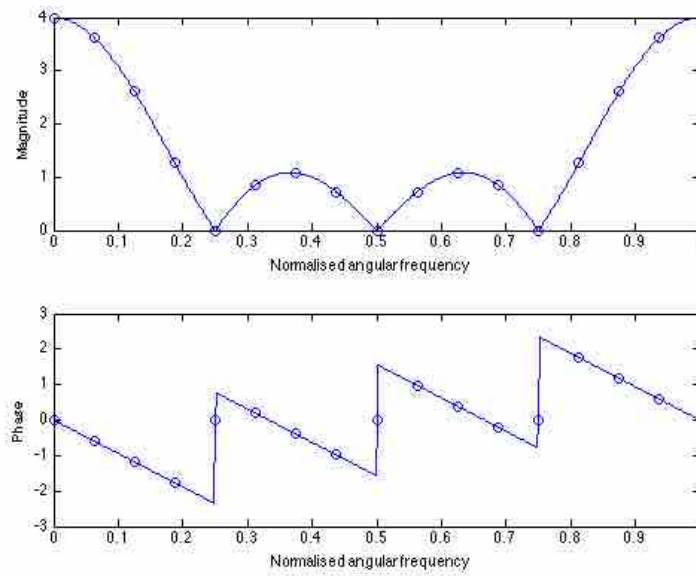
Παράδειγμα DFT (N=4):  $x(n) = \delta(n) + \delta(n-1) + \delta(n-2) + \delta(n-3)$



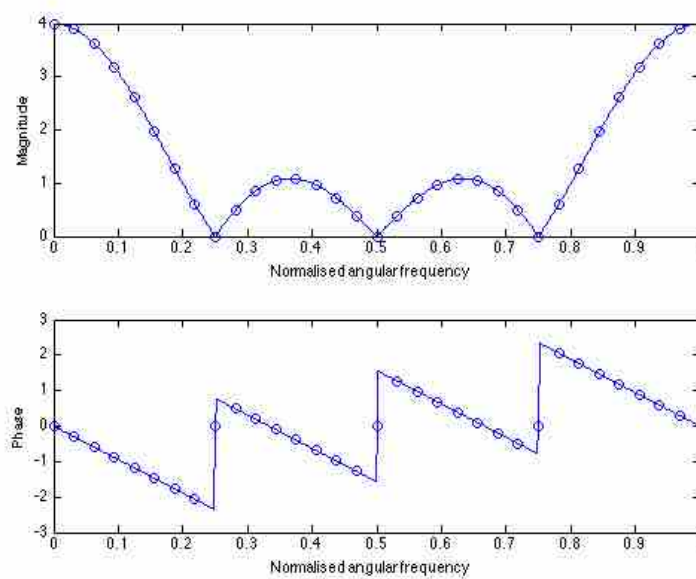
Παράδειγμα DFT (N=8):  $x(n) = \delta(n) + \delta(n-1) + \delta(n-2) + \delta(n-3)$



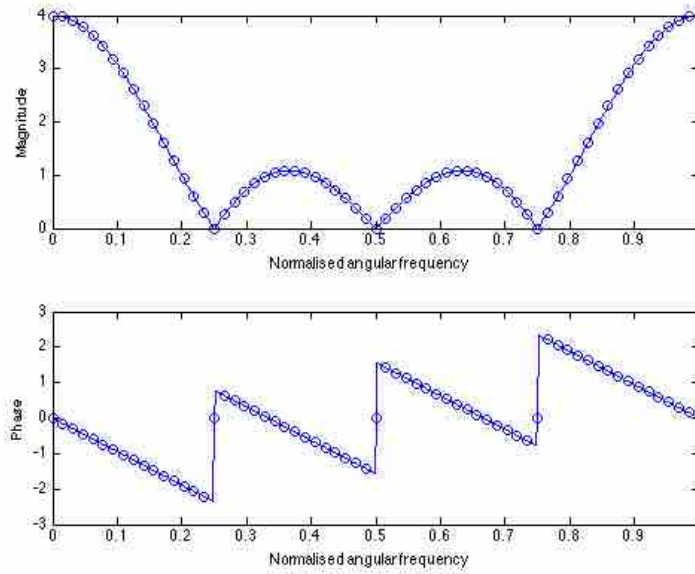
Παράδειγμα DFT (N=16):  $x(n) = \delta(n) + \delta(n-1) + \delta(n-2) + \delta(n-3)$



Παράδειγμα DFT (N=32):  $x(n) = \delta(n) + \delta(n-1) + \delta(n-2) + \delta(n-3)$



Παράδειγμα DFT (N=64):  $x(n) = \delta(n) + \delta(n-1) + \delta(n-2) + \delta(n-3)$



Example

$\{1, 1, 0, 0\}$

$x(0) = 1, x(1) = 1, x(2) = 0, x(3) = 0$

$$X(0) = \sum_{n=0}^3 x(n)W_4^{n \cdot 0} = x(0)W_4^{0 \cdot 0} + x(1)W_4^{1 \cdot 0} + x(2)W_4^{2 \cdot 0} + x(3)W_4^{3 \cdot 0} =$$

$$= x(0) + x(1) + x(2) + x(3) = 1 + 1 + 0 + 0 = 2$$

$$X(1) = \sum_{n=0}^3 x(n)W_4^{n \cdot 1} = x(0)W_4^{0 \cdot 1} + x(1)W_4^{1 \cdot 1} + x(2)W_4^{2 \cdot 1} + x(3)W_4^{3 \cdot 1} =$$

$$= x(0) + x(1) \cdot (-j) + x(2) \cdot (-1) + x(3) \cdot j = 1 + 1 \cdot (-j) + 0 \cdot (-1) + 0 \cdot j = 1 - j$$

where

$$W_4^1 = e^{-j\frac{2\pi}{4}} = e^{-j\frac{\pi}{2}} = \cos\frac{\pi}{2} - j\sin\frac{\pi}{2} = 0 - j = -j$$

$$W_4^2 = e^{-j\frac{2\pi \cdot 2}{4}} = e^{-j\pi} = \cos\pi - j\sin\pi = -1$$

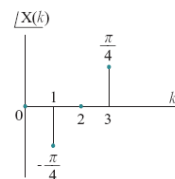
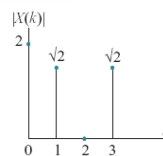
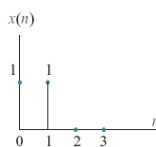
$$W_4^3 = e^{-j\frac{2\pi \cdot 3}{4}} = e^{-j\frac{3\pi}{2}} = \cos\frac{3\pi}{2} - j\sin\frac{3\pi}{2} = 0 - j(-1) = j$$

$X(2) = 0$

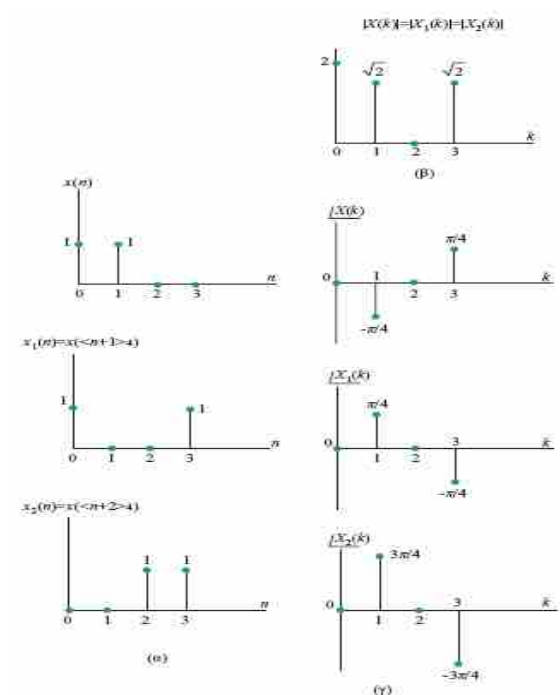
$X(3) = 1 + j$

$\{2, 1-j, 0, 1+j\}$

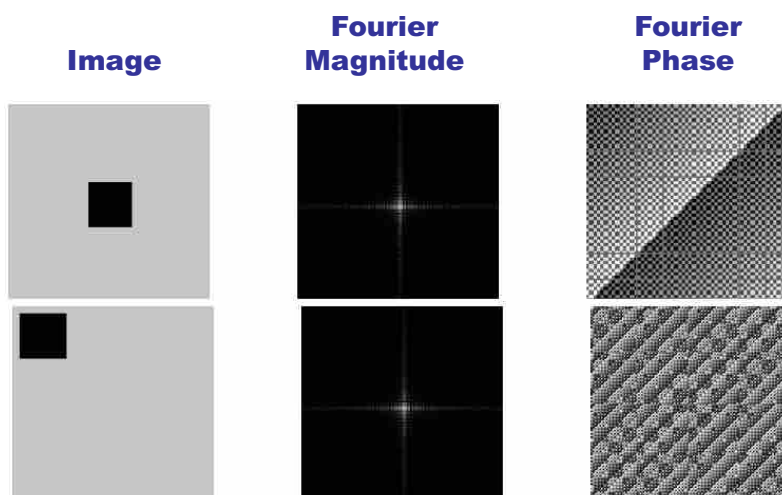
or  $\left\{ 2, \sqrt{2}e^{-j\frac{\pi}{4}}, 0, \sqrt{2}e^{j\frac{\pi}{4}} \right\}$



## Time shifting affects phase only!



## Time shifting affects phase only!





## FFT Interactive Demo: 2D Case

The screenshot shows a web-based interface for a 2D FFT demo. At the top, there is a 'Load Image' button and an 'Input Image' field containing 'cin1.gif', with a status message 'Image loaded successfully'. Below this is a 'Frequency Domain Operators' section with several controls: 'Remove Area' and 'Keep Area' buttons, coordinate fields for X1 (10), Y1 (10), X2 (20), and Y2 (20); 'Apply Notch Filter' with 'Width' (5) and 'Radius' (15); 'Apply Frequency Filter' with a dropdown set to 'High Pass Filter' and 'Radius' (50); and 'Gaussian Smooth' with 'Size' (30) and X/Y coordinates (0, 0). An 'Apply' section contains 'Fourier Transform' and 'Inverse Fourier Transform' buttons, and a 'Time Taken (msecs):' field showing '1419'. The main area is divided into 'Input Image' and 'Output Image' sections, each showing a 256 x 256 pixel image of a person's face. To the right is a plot area with a 'Reset Images' button. The plot has a vertical axis from -128 to 127 and a horizontal axis from -128 to 127. A legend on the left of the plot lists: 'Magnitude', 'Magnitude Log', 'Phase Angle', 'Real', 'Real Log', 'Imaginary', and 'Imaginary Log'. The plot itself shows a central bright spot with a horizontal line of interference.

<http://homepages.inf.ed.ac.uk/rbf/HIPR2/fftdemo.htm>

## FFT Interactive Demo: 2D Case

The screenshot shows a Java-based '2D Fourier Applet' interface. On the left is a red-tinted portrait of a man. To its right is a 'Tool' section with icons for a hand, a selection tool, and a zoom tool. Below the tools are 'Data Display' and 'FFT Display' sections, each with a 'Brightness' slider and 'Discard Info' buttons. The 'FFT Display' shows a colorful 2D frequency spectrum. At the bottom, there are several small thumbnail images of different faces and patterns. The interface is designed for interactive exploration of 2D Fourier transforms.

<http://www.brainflux.org/java/classes/FFT2DApplet.html>

## Υπολογισμός Αντίστροφου DFT

$$\text{IDFT: } x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) W_N^{-nk}, \quad n=0,1,\dots,N-1 \quad (1)$$

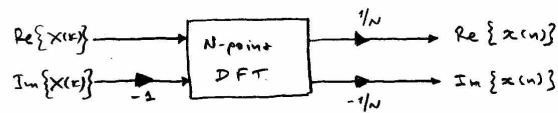
Πολλαπλασιάζοντας και τα δύο μέλη με  $N$  και λαμβάνοντας τον συζυγή μιγαδικό, έχουμε:

$$N x^*(n) = \underbrace{\sum_{k=0}^{N-1} X^*(k) W_N^{nk}}_{\text{DFT}} \quad (2)$$

Το δεξιό μέλος είναι ο DFT της κυκλωθείσας  $X^*(k)$ , και μπορεί να υπολογιστεί (όπου που FFT). Άρα, ο γινόμενος IDFT υπολογίζεται ως:

$$x(n) = \frac{1}{N} \left\{ \sum_{k=0}^{N-1} X^*(k) W_N^{nk} \right\}^* \quad (3)$$

Διαγραμματικά αυτό φαίνεται στο παρακάτω σχήμα.



# ΚΥΚΛΙΚΗ ΟΛΙΣΘΗΣΗ

**Circular shift of a sequence of finite length is equivalent to the linear shift of its periodic extension**

$$x_p(n) = \sum_{q=-\infty}^{\infty} x(n - qN)$$

$$x'_p(n) = x_p(n - n_0) = \sum_{q=-\infty}^{\infty} x(n - n_0 - qN)$$

$$x'(n) = x(\langle n - n_0 \rangle_N)$$

$$x(\langle n - n_0 \rangle_N) = \begin{cases} x(n - n_0 + N) & \text{για } n = 0, 1, \dots, n_0 - 1 \\ x(n - n_0) & \text{για } n = n_0, \dots, N - 1 \end{cases}$$

$$x'(0) = x(\langle 0 - 2 \rangle_6) = x(0 - 2 + 6) = x(4) = 5$$

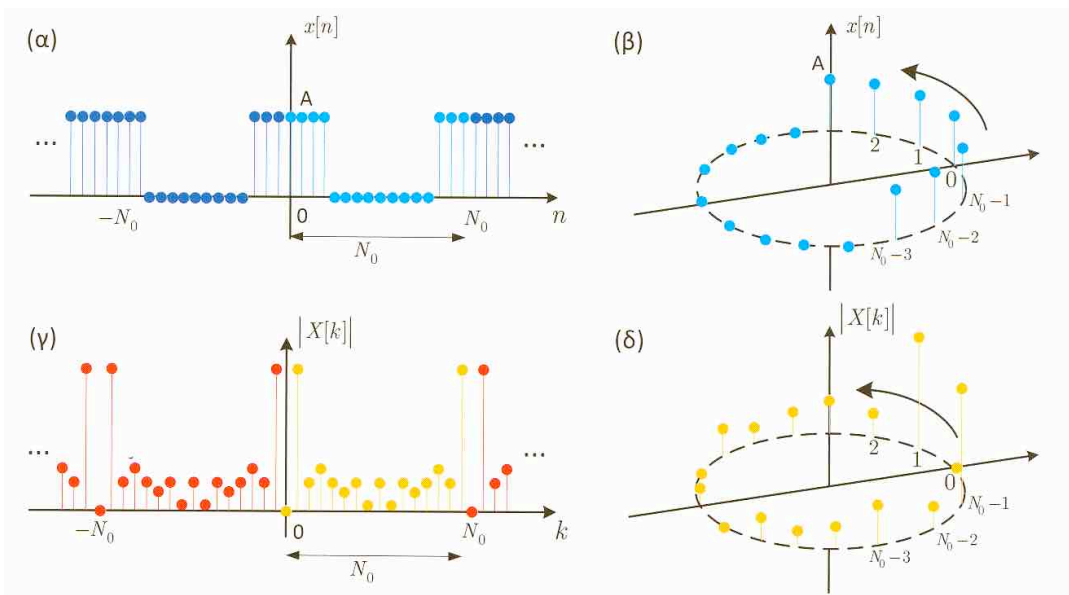
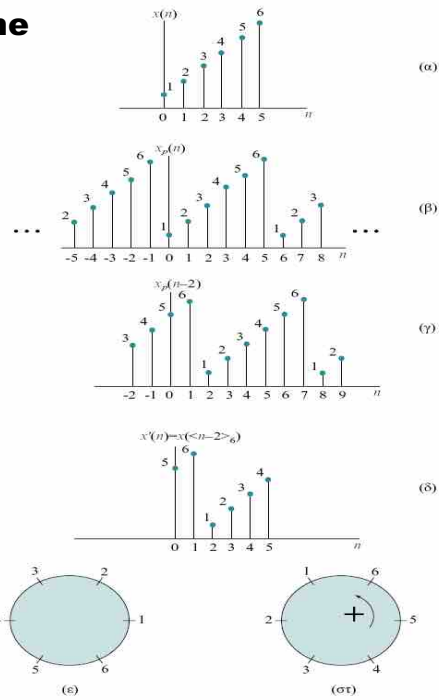
$$x'(1) = x(\langle 1 - 2 \rangle_6) = x(1 - 2 + 6) = x(5) = 6$$

$$x'(2) = x(\langle 2 - 2 \rangle_6) = x(0) = 1$$

$$x'(3) = x(\langle 3 - 2 \rangle_6) = x(1) = 2$$

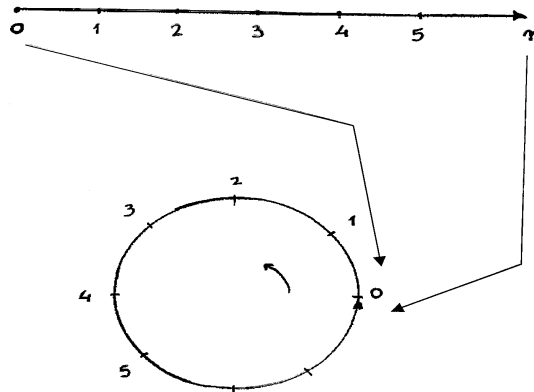
$$x'(4) = x(\langle 4 - 2 \rangle_6) = x(2) = 3$$

$$x'(5) = x(\langle 5 - 2 \rangle_6) = x(3) = 4$$



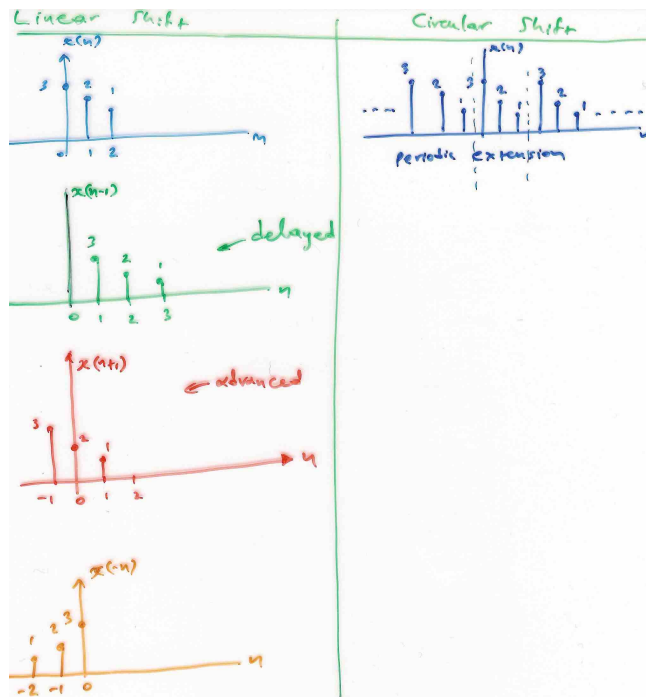
ΠΗΓΗ: Βιβλίο Γ. Π. Καφεντζή: "Επεξεργασία Σήματος Συνεχούς και Διακριτού Χρόνου - Μια πρώτη εισαγωγή", Εκδόσεις Gutenberg, 2019

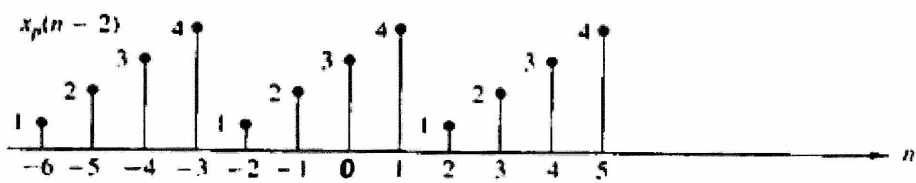
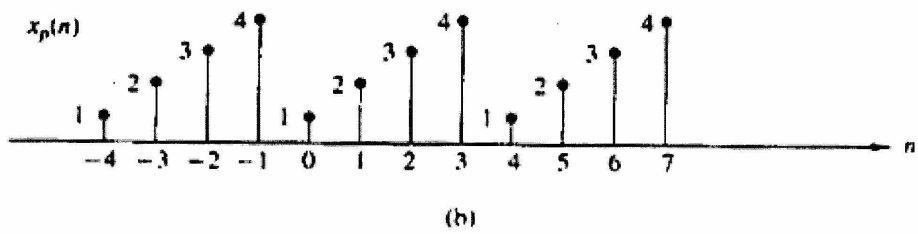
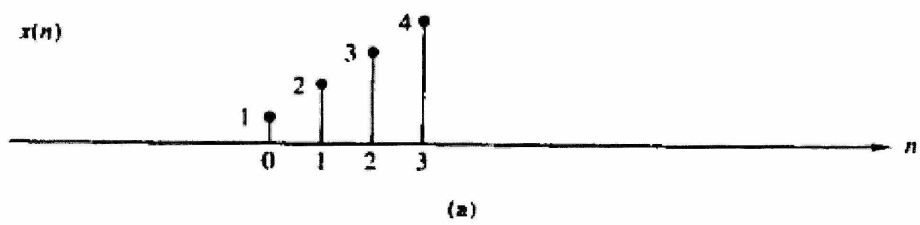
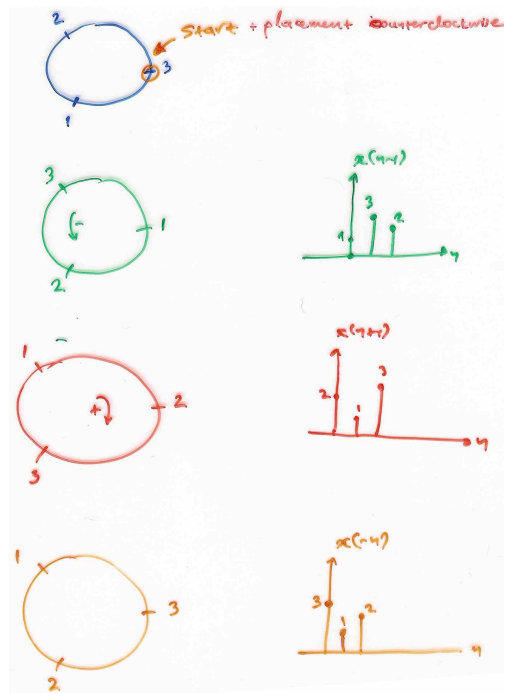
## Circular Shift: Which Direction?

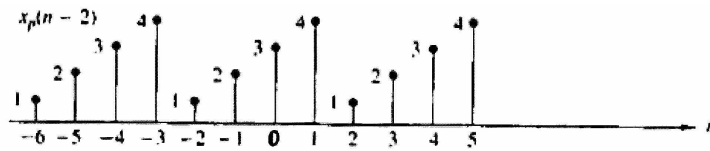


**Counterclockwise for positive  $n$ .**

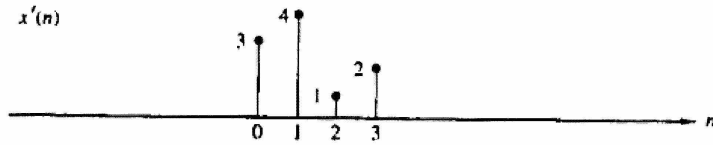
Shifting to the right (i.e. towards positive  $n$ ) corresponds to counterclockwise rotation.



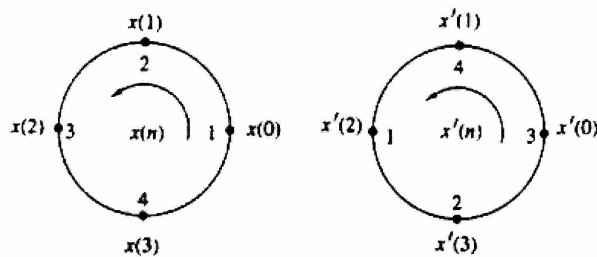




(c)



(d)



ΚΥΚΛΙΚΕΣ ΣΥΜΜΕΤΡΙΕΣ ΣΗΜΑΤΟΣ

- Ο υπολογισμός του DFT  $N$ -εμφάνων ενός σήματος  $x(n)$  πεπερασμένου διάρκειας  $L \leq N$  ισοδυναμεί με τον υπολογισμό του DFT  $N$ -εμφάνων ενός περιοδικού σήματος  $x_p(n)$  περιόδου  $N$ , το οποίο έχει προκύψει από την περιοδική επέκταση του  $x(n)$ , δηλαδή

$$x_p(n) = \sum_{q=-\infty}^{\infty} x(n - qN)$$

Ας υποθέσουμε ότι το περιοδικό σήμα  $x_p(n)$  υφίσταται ολίσθηση προς τα δεξιά κατά  $k$  μονάδες. Το σήμα που προκύπτει είναι το  $x_p(n-k)$ , όπου

$$x_p(n-k) = \sum_{q=-\infty}^{\infty} x(n-k-qN)$$

Το πεπερασμένο διάρκειας σήμα  $x'(n)$  που αποτελείται από τα  $N$  δείγματα μεταξύ 0 και  $N-1$ , δηλ.

$$x'(n) = \begin{cases} x_p(n-k) & \text{για } 0 \leq n \leq N-1 \\ 0 & \text{αλλιώς} \end{cases}$$

προκύπτει από το αρχικό σήμα  $x(n)$  μέσω κυκλικής ολίσθησης. (βλ. σήμα στην εφεξής σειρά για  $N=4$ ).

Συμπέρασμα: Η κυκλική ολίσθηση μιας ακολουθίας  $N$ -εμφάνων ισοδυναμεί με τη γραμμική ολίσθηση (εταπόνηση) της περιοδικής της επέκτασης.

- Μια ακολουθία  $N$ -επιπέδων ονομάζεται κυκλικά άρτια εάν είναι συσφαιρική γύρω από το επίπεδο 0 (μηδέν) του κύκλου:

$$x(N-n) = x(n) \quad 1 \leq n \leq N-1$$

- Μια ακολουθία  $N$ -επιπέδων ονομάζεται κυκλικά περιττή εάν είναι αντισφαιρική γύρω από το επίπεδο 0 (μηδέν) του κύκλου:

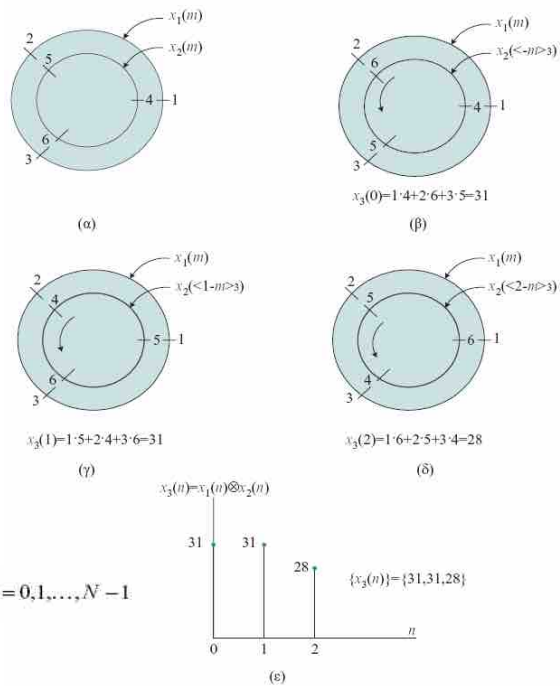
$$x(N-n) = -x(n) \quad 1 \leq n \leq N-1$$

- Η χρονική αναστροφή (time reversal) και ακολουθίας  $N$ -επιπέδων προκύπτει από την αναστροφή των στοιχείων της γύρω από το επίπεδο 0 (μηδέν) του κύκλου:

$$x\langle -n \rangle_N = x(N-n) \quad 0 \leq n \leq N-1$$

Η χρονική αναστροφή ισοδυναμεί με τη σχεδίαση της  $x(n)$  κατά τη φορά των δεικτών του ρολογαϊού πάνω στον κύκλο.

## Circular Convolution



$$x_1(n) \otimes x_2(n) = \sum_{m=0}^{N-1} x_1(m)x_2\langle n-m \rangle_N, \quad n = 0, 1, \dots, N-1$$

## CONVOLUTION

● Linear Convolution:  $x_1(n) * x_2(n) = \sum_{l=-\infty}^{\infty} x_1(l) x_2(n-l) = \sum_{l=-\infty}^{\infty} x_1(n-l) x_2(l)$

Note: The linear convolution of two sequences of lengths  $N_1$  and  $N_2$  will result in a sequence of length  $N_1 + N_2 - 1$ .

● Circular Convolution:  $x_1(n) \otimes x_2(n) = \sum_{l=0}^{N-1} x_1(l) x_2(n-l) = \sum_{l=0}^{N-1} x_1(n-l) x_2(l)$

Note: An equal number of samples are required to perform circular convolution.

- ▶ Multiplication of two DFTs corresponds to a circular convolution of their time-domain sequences.
- ▶ In order to implement a FIR filter, a linear convolution is required.
- ▶ The procedures for converting a circular convolution into a linear convolution are the OVERLAP-ADD and OVERLAP-SAVE fast convolution algorithms.

**ΑΣΚΗΣΗ** Να υπολογιστεί η κυκλική συνέλιξη των ακολουθιών  $x_1(n) = \{1, 2, 3\}$  και  $x_2(n) = \{4, 5, 6\}$  μέσω του DFT και IDFT.

**ΛΥΣΗ** Κατ' αρχήν υπολογίζουμε τον DFT των ακολουθιών  $x_1(n)$  και  $x_2(n)$  για  $N=3$ .

$$X_1(k) = \sum_{n=0}^2 x_1(n) e^{-j\frac{2\pi}{3}nk} = x_1(0) + x_1(1) e^{-j\frac{2\pi}{3}k} + x_1(2) e^{-j\frac{2\pi}{3}2k} = 1 + 2 e^{-j\frac{2\pi}{3}k} + 3 e^{-j\frac{4\pi}{3}k}$$

$$X_2(k) = \sum_{n=0}^2 x_2(n) e^{-j\frac{2\pi}{3}nk} = 4 + 5 e^{-j\frac{2\pi}{3}k} + 6 e^{-j\frac{4\pi}{3}k}$$

$$X_1(0) = 6$$

$$X_2(0) = 15$$

$$X_1(1) = -\frac{3}{2} + j\frac{\sqrt{3}}{2}$$

$$X_2(1) = -\frac{3}{2} + j\frac{\sqrt{3}}{2}$$

$$X_1(2) = -\frac{3}{2} - j\frac{\sqrt{3}}{2}$$

$$X_2(2) = -\frac{3}{2} - j\frac{\sqrt{3}}{2}$$

Πολλαπλασιάζοντας (στοιχείο προς στοιχείο) τις ακολουθίες  $X_1(k)$  και  $X_2(k)$  παίρνουμε την «ακολουθία»

$$X_3(k) = X_1(k) \cdot X_2(k)$$

οπότε  $X_3(0) = 90$

$$X_3(1) = \frac{9}{2} - j\frac{3\sqrt{3}}{2}$$

$$X_3(2) = \frac{9}{2} + j\frac{3\sqrt{3}}{2}$$



Ο αντίστροφος DFT (IDFT) της  $X_3(k)$  θα μας δώσει την ακολουθία  $x_3(n)$  που δεν είναι άλλη από την κυματική συνάρτηση των  $x_1(n)$  και  $x_2(n)$ .

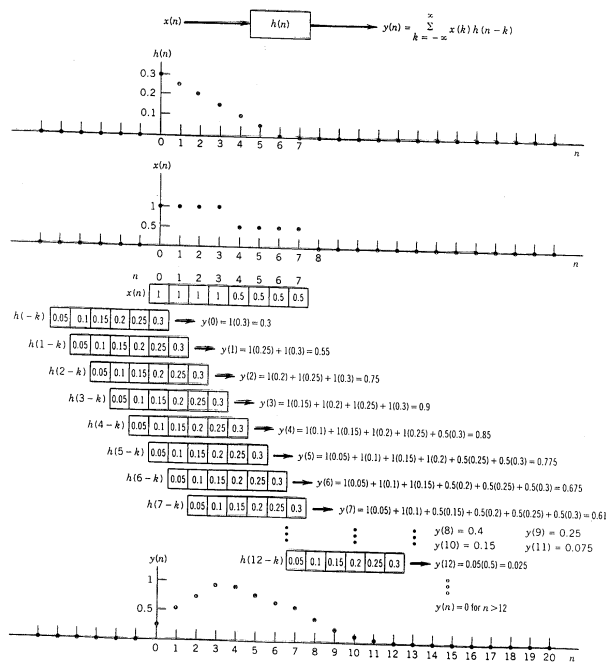
$$x_3(n) = \frac{1}{3} \sum_{k=0}^2 X_3(k) e^{j \frac{2\pi}{3} nk} = \frac{1}{3} \left[ X_3(0) + X_3(1) e^{j \frac{2\pi}{3} n} + X_3(2) e^{j \frac{4\pi}{3} n} \right] = \frac{1}{3} \left[ 90 + \left( \frac{3}{2} - j \frac{3\sqrt{3}}{2} \right) e^{j \frac{2\pi}{3} n} + \left( \frac{3}{2} + j \frac{3\sqrt{3}}{2} \right) e^{j \frac{4\pi}{3} n} \right]$$

Τελικά, για  $n=0, 1, 2$  βρίσκουμε αντίστοιχα:

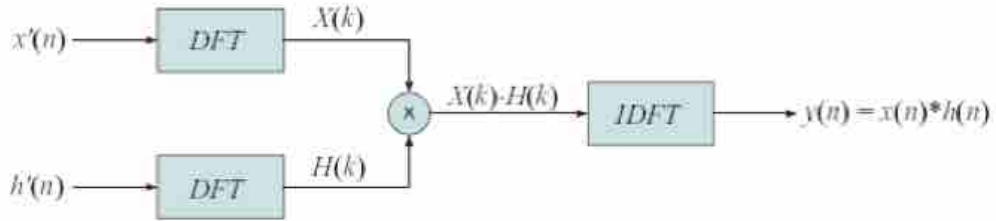
$$x_3(0) = 31 \quad x_3(1) = 31 \quad x_3(2) = 28$$

$$x_3(n) = \{31, 31, 28\}$$

## Linear Convolution Example



## Linear Convolution via the DFT



Πλήθος στοιχείων κάθε ακολουθίας	Απευθείας υπολογισμός της γραμμικής συνέλιξης στο πεδίο του χρόνου	Υπολογισμός της γραμμικής συνέλιξης στο πεδίο της συχνότητας μέσω του FFT
$(N)$	$(N^2)$	$(12N \log_2 2N + 8N)$
4	16	176
8	64	448
16	256	1088
32	1024	2560
64	4096	5888
128	16384	13312
256	65536	29696
512	262144	65536
1024	1048576	143360

## DFT

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j\frac{2\pi}{N}kn}, \quad k=0,1,\dots,N-1$$

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j\frac{2\pi}{N}kn}, \quad n=0,1,\dots,N-1$$

← ΓΩΩJ f-ετακ. (DFT)

← αντιστροφας f-ετακ. (IDFT)

Ορισοντας με  $W_N = e^{-j\frac{2\pi}{N}}$  τιν Ν-οστη ριζα τινς μοναδα, οι παρανω εκφασ εκφραζονται ωσ εξησ:

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn}, \quad k=0,1,\dots,N-1$$

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) W_N^{-kn}, \quad n=0,1,\dots,N-1$$

← DFT

← IDFT

The DFT in matrix format

$$\mathbf{x} = [x(0) \ x(1) \ \dots \ x(N-1)]^T$$

$$\mathbf{X} = [X(0) \ X(1) \ \dots \ X(N-1)]^T$$

**$W_N = e^{-j2\pi/N}$  is the Nth root of unity**

$$\mathbf{W} = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & W_N & W_N^2 & \dots & W_N^{N-1} \\ 1 & W_N^2 & W_N^4 & \dots & W_N^{2(N-1)} \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ 1 & W_N^{N-1} & W_N^{2(N-1)} & \dots & W_N^{(N-1)(N-1)} \end{bmatrix}$$

	n →							
	0	1	2	3	4	5	6	7
0	1	1	1	1	1	1	1	1
1	1	a	-j	-b	-1	-a	j	b
2	1	-j	-1	j	1	-j	-1	j
3	1	-b	j	a	-1	b	-j	-a
4	1	-1	1	-1	1	-1	1	-1
5	1	-a	-j	b	-1	a	j	-b
6	1	j	-1	-j	1	j	-1	-j
7	1	b	j	-a	-1	-b	-j	a

$$\mathbf{X} = \mathbf{W}\mathbf{x}$$

$$\mathbf{x} = \mathbf{W}^{-1}\mathbf{X}$$

$$a = \frac{1-j}{\sqrt{2}}, \quad b = \frac{1+j}{\sqrt{2}}, \quad j = \sqrt{-1}$$

## Discrete Fourier Transform visualized

$$\begin{pmatrix} \text{⊕} & \text{⊕} & \text{⊕} & \text{⊕} & \text{⊕} & \text{⊕} & \text{⊕} & \text{⊕} \\ \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} \\ \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} \\ \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} \\ \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} \\ \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} \\ \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} \\ \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} \end{pmatrix} \cdot \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{pmatrix} = \begin{pmatrix} X_0 \\ X_1 \\ X_2 \\ X_3 \\ X_4 \\ X_5 \\ X_6 \\ X_7 \end{pmatrix}$$

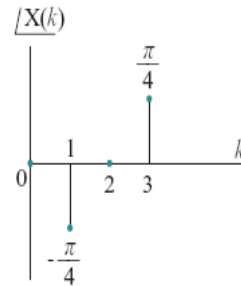
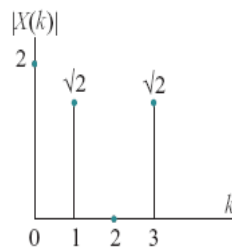
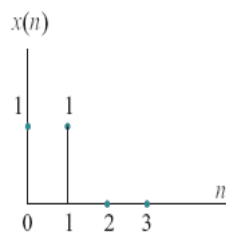
## Inverse DFT visualized

$$\frac{1}{8} \cdot \begin{pmatrix} \text{⊕} & \text{⊕} & \text{⊕} & \text{⊕} & \text{⊕} & \text{⊕} & \text{⊕} & \text{⊕} \\ \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} \\ \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} \\ \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} \\ \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} \\ \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} \\ \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} \\ \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} & \text{⊕} & \text{⊗} \end{pmatrix} \cdot \begin{pmatrix} X_0 \\ X_1 \\ X_2 \\ X_3 \\ X_4 \\ X_5 \\ X_6 \\ X_7 \end{pmatrix} = \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{pmatrix}$$

Example  $x(n) = \{1, 1, 0, 0\}$

$$W_4 = \begin{bmatrix} W_4^0 & W_4^0 & W_4^0 & W_4^0 \\ W_4^0 & W_4^1 & W_4^2 & W_4^3 \\ W_4^0 & W_4^2 & W_4^4 & W_4^6 \\ W_4^0 & W_4^3 & W_4^6 & W_4^9 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & W_4^1 & W_4^2 & W_4^3 \\ 1 & W_4^2 & W_4^4 & W_4^6 \\ 1 & W_4^3 & W_4^6 & W_4^9 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix}$$

$$X = Wx = \begin{bmatrix} 2 \\ 1-j \\ 0 \\ 1+j \end{bmatrix}$$



ΑΙΤΗΣΗ

Να αποδειχθεί ότι ο απευθείας υπολογισμός ενός  $N$ -συντελών DFT απαιτεί  $4N^2$  πραγματικούς ποσ/φοίς και  $(4N-2)N$  πραγματικές προσδέσεις.

ΛΥΣΗ

Θέλουμε να υπολογίσουμε την DFT:  $X(k) = \sum_{n=0}^{N-1} x(n) W_N^{nk} \quad k=0,1,2,\dots,N-1$

Ο υπολογισμός των  $N$  συντελεστών  $X(k)$  απαιτεί  $N^2$  φυσικούς ποσ/φοίς και  $N(N-1)$  φυσικές προσδέσεις.

Όπως κάθε φυσικός ποσ/φοίς απαιτεί 4 πραγμ. ποσ/φοίς και 2 πραγματικές προσδέσεις, ενώ κάθε φυσική προσδέση απαιτεί 2 πραγματικές προσδέσεις. Άρα έχουμε:

$$O_M = 4 \cdot N^2$$

$$O_A = \underbrace{N(N-1)}_{\text{πρόσδεσες}} \cdot 2 + \underbrace{N^2}_{\text{ποσ/φοίς}} \cdot 2 = 2N^2 - 2N + 2N^2 = 4N^2 - 2N = (4N-2)N$$

# Fourier Transform

## Fast Fourier Transform

discovered by Carl Gauss ~1805  
re-invented by Cooley & Tukey in 1965

wikipedia

Joseph Fourier



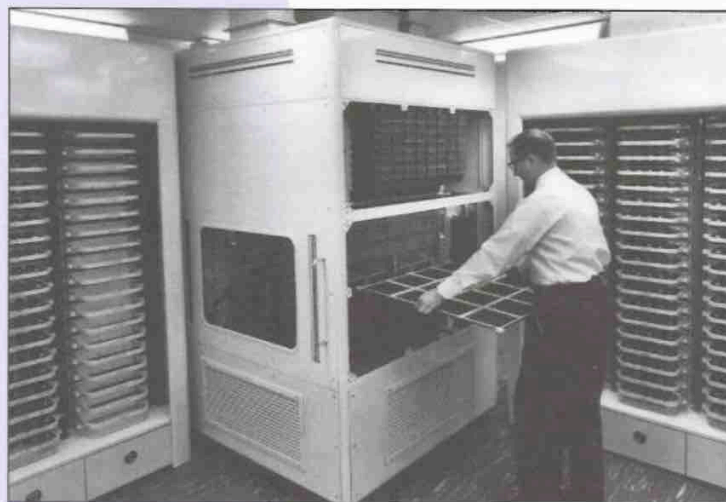
Jean Baptiste Joseph Fourier

<b>Born</b>	March 21, 1768 Auxerre, Yonne, France
<b>Died</b>	May 16, 1830 (aged 62) Paris, France
<b>Residence</b>	<span><span><span></span></span><span> </span></span> France
<b>Nationality</b>	<span><span><span></span></span><span> </span></span> French
<b>Fields</b>	Mathematician, physicist, and historian
<b>Institutions</b>	École Normale École Polytechnique
<b>Alma mater</b>	École Normale
<b>Doctoral advisor</b>	Joseph Lagrange
<b>Doctoral students</b>	Gustav Dirichlet Giovanni Plana Claude-Louis Navier
<b>Known for</b>	Fourier transform
<b>Religious stance</b>	Roman Catholic

### Top 10 Algorithms of the 20th Century

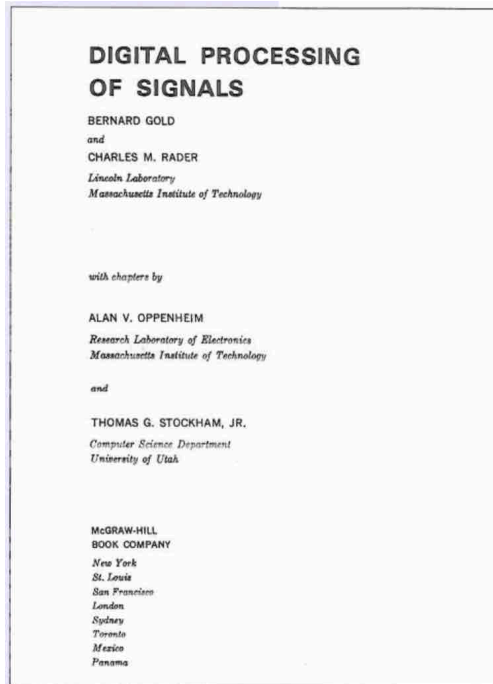
- 1946: The Monte Carlo method.
- 1947: Simplex Method for Linear Programming.
- 1950: Krylov Subspace Iteration Method.
- 1951: The Decompositional Approach to Matrix Computations.
- 1957: The Fortran Compiler.
- 1959: QR Algorithm for Computing Eigenvalues.
- 1962: Quicksort Algorithms for Sorting.
- 1965: Fast Fourier Transform.
- 1977: Integer Relation Detection.
- 1987: Fast Multipole Method.

Dongarra & Sullivan, IEEE Comput. Sci. Eng., Vol. 2(1):22–23 (2000).



*This is the Lincoln Lab TX-2 computer, which Ben Gold and Charles Rader used to perform numerical simulations of the performance of analog filters. The photograph shows Don Ellis removing one bit-plane from the 64K-word ferrite-core memory. (Lincoln Lab photo reproduced by permission.)*

### 1961 introduction of stereo FM broadcasting



*This is the title page of Gold and Rader's Digital Processing of Signals (1969), which was the first textbook on digital signal processing. (Reproduced by permission of McGraw Hill.)*

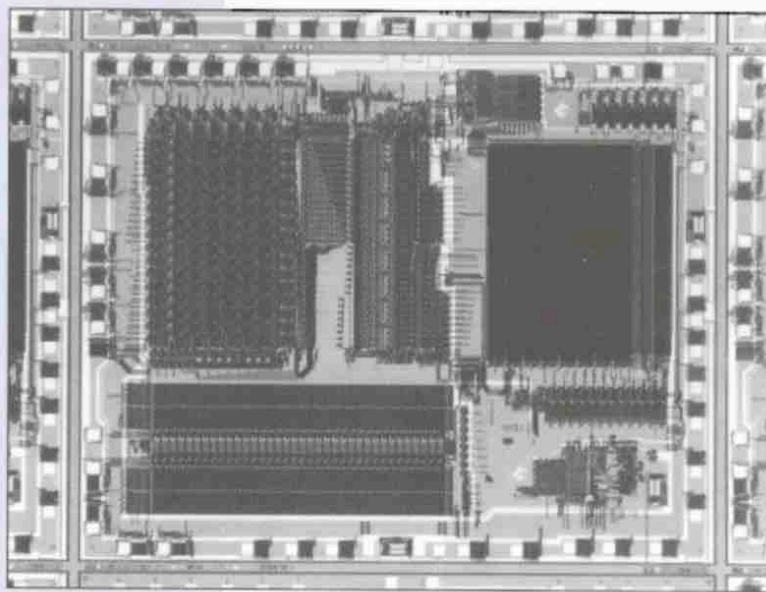


*This photograph shows the Lincoln Fast Digital Processor (FDP), built under Charles Rader's direction at Lincoln Lab. (Lincoln Lab photo reproduced by permission.)*

The 1970s were a watershed in the history of signal processing for the development of signal-processing hardware. The decade opened with the completion of what may be the first real-time DSP computer: the Lincoln Fast Digital Processor (FDP). A special-purpose, parallel-processing computer, the FDP performed signal-processing tasks about a hundred times as fast as the general-purpose computers of the time. There followed in 1974 the Lincoln Digital Voice Terminal, a computer built to carry out a variety of speech-compression algorithms. A more powerful version, the Lincoln Digital Signal Processor (LDSP), was built in four copies and remained the main research tool for the Signal Processing Group at Lincoln Lab into the 1980s.

1971

15 November: Intel announces the microprocessor (the 4004)



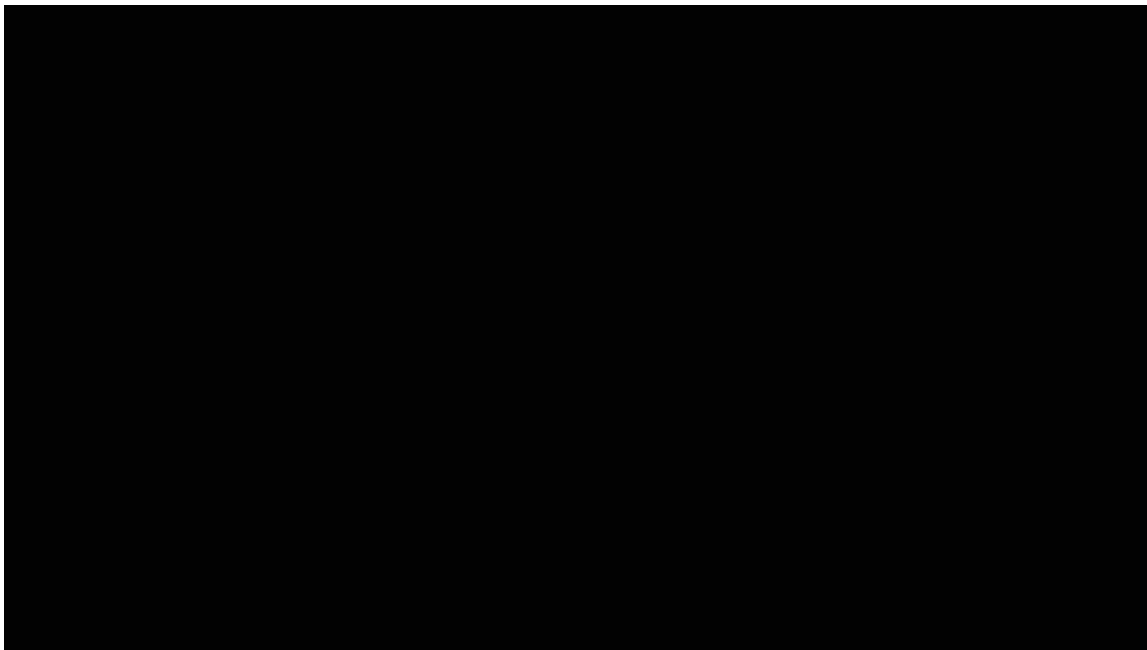
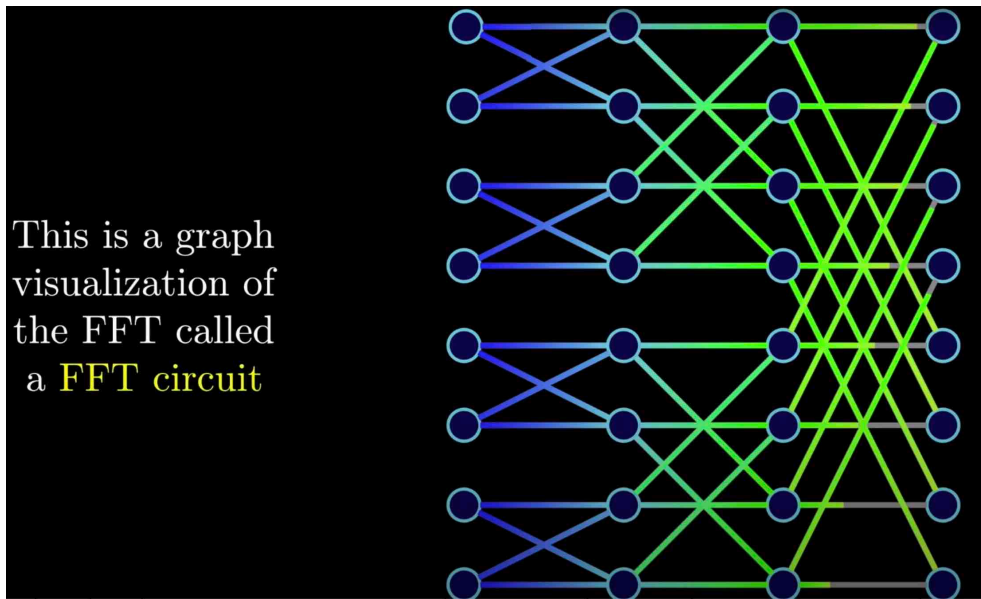
*This was the first single-chip DSP manufactured by Texas Instruments.  
(Texas instruments photo reproduced with permission.)*

Source: <https://signalprocessingsociety.org/uploads/history/history.pdf>

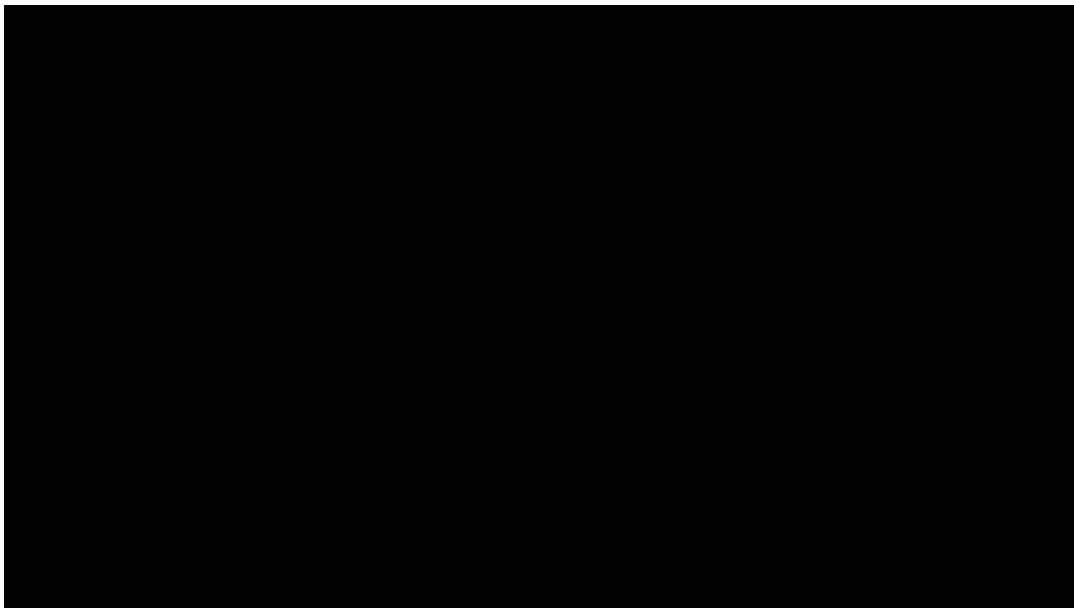
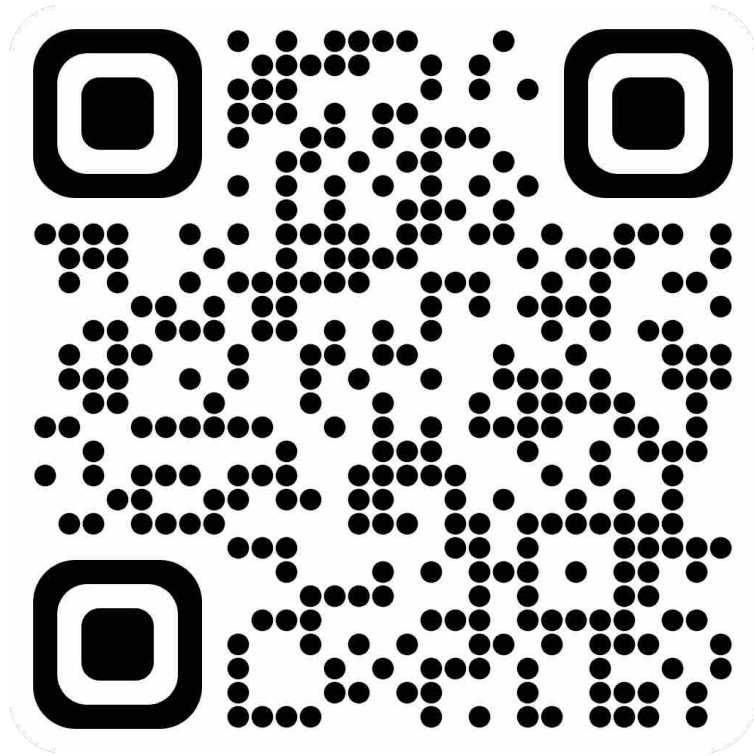
## 1982 Sony introduces the 3.5-inch diskette

# FFT

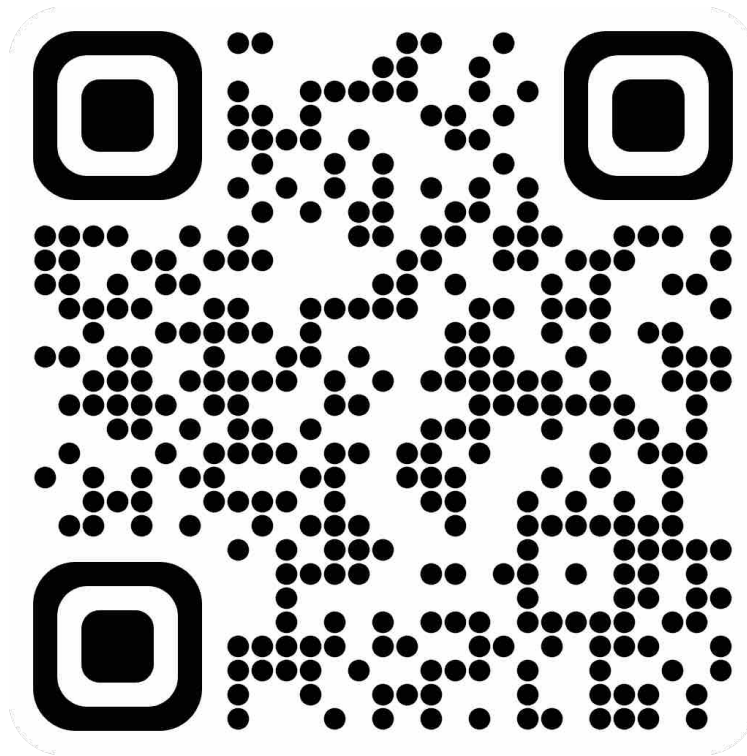




<https://youtu.be/nmgFG7PUHfo?t=929>



<https://youtu.be/h7apO7q16V0?t=52>



[https://www.youtube.com/watch?v=EsJGul7e\\_ZQ](https://www.youtube.com/watch?v=EsJGul7e_ZQ)

## Fast Fourier Transform (FFT)

$$\begin{aligned}
 X(k) &= \sum_{n=0}^{N-1} x(n)W_N^{nk} = \sum_{n=0}^{\frac{N}{2}-1} x(2n)W_N^{2nk} + \sum_{n=0}^{\frac{N}{2}-1} x(2n+1)W_N^{(2n+1)k} = \\
 &= \sum_{n=0}^{\frac{N}{2}-1} x(2n)W_N^{2nk} + W_N^k \sum_{n=0}^{\frac{N}{2}-1} x(2n+1)W_N^{2nk}
 \end{aligned}$$

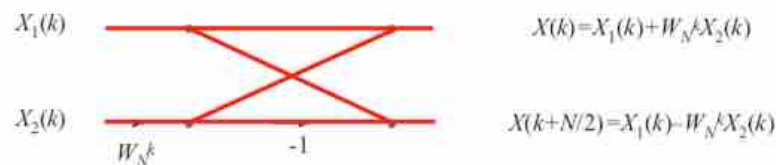
$$W_N^{2nk} = e^{-j\frac{2\pi}{N}2nk} = e^{-j\frac{2\pi}{N/2}nk} = W_{N/2}^{nk}$$

thus 
$$X(k) = \underbrace{\sum_{n=0}^{\frac{N}{2}-1} x(2n)W_{N/2}^{nk}}_{X_1(k)} + W_N^k \underbrace{\sum_{n=0}^{\frac{N}{2}-1} x(2n+1)W_{N/2}^{nk}}_{X_2(k)}$$

or 
$$X(k) = X_1(k) + W_N^k X_2(k)$$

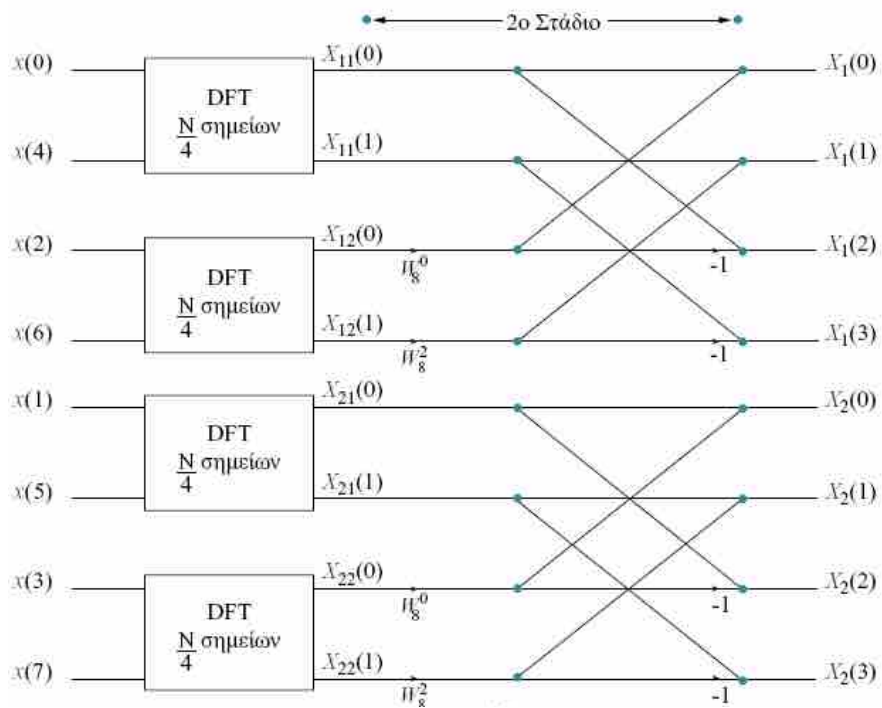
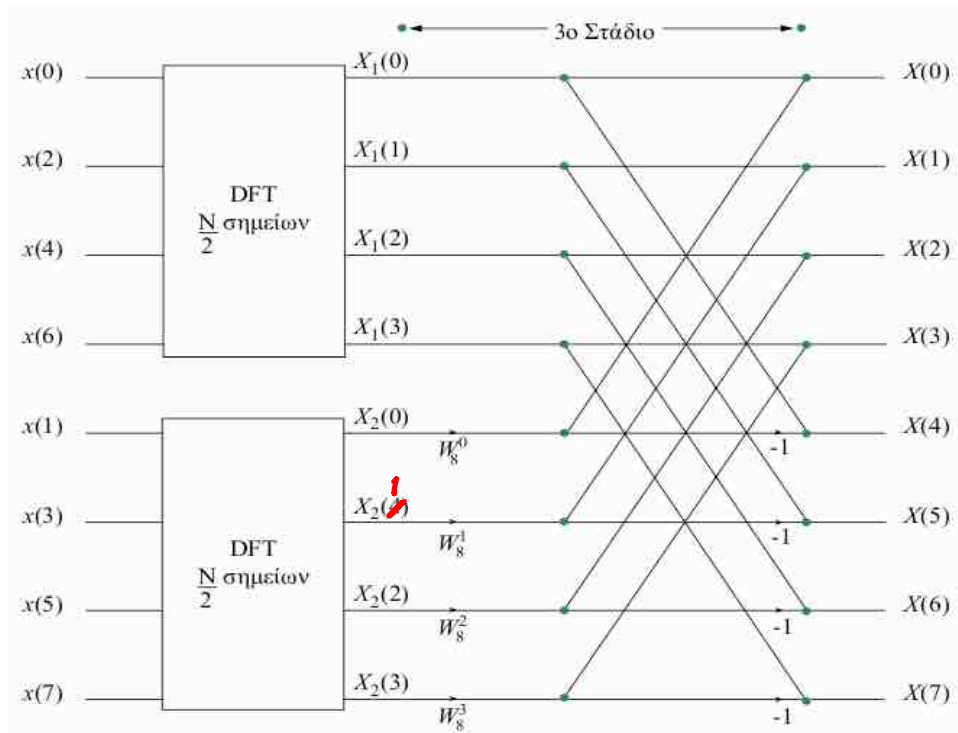
but 
$$W_N^{k+\frac{N}{2}} = -W_N^k \quad W_N^{N\left(k+\frac{N}{2}\right)} = W_N^{nk} \quad X\left(k+\frac{N}{2}\right) = X_1(k) - W_N^k X_2(k)$$

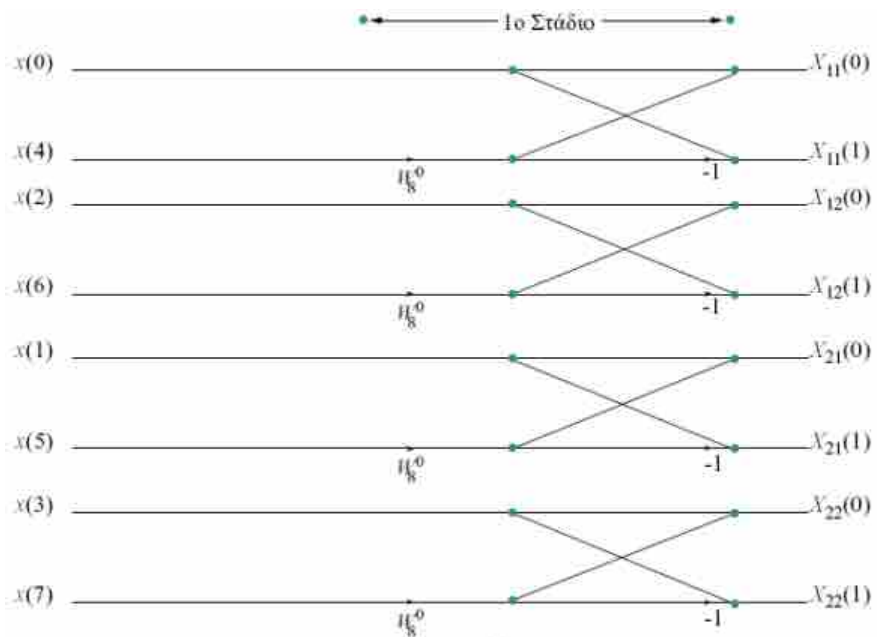
## Fast Fourier Transform (FFT)



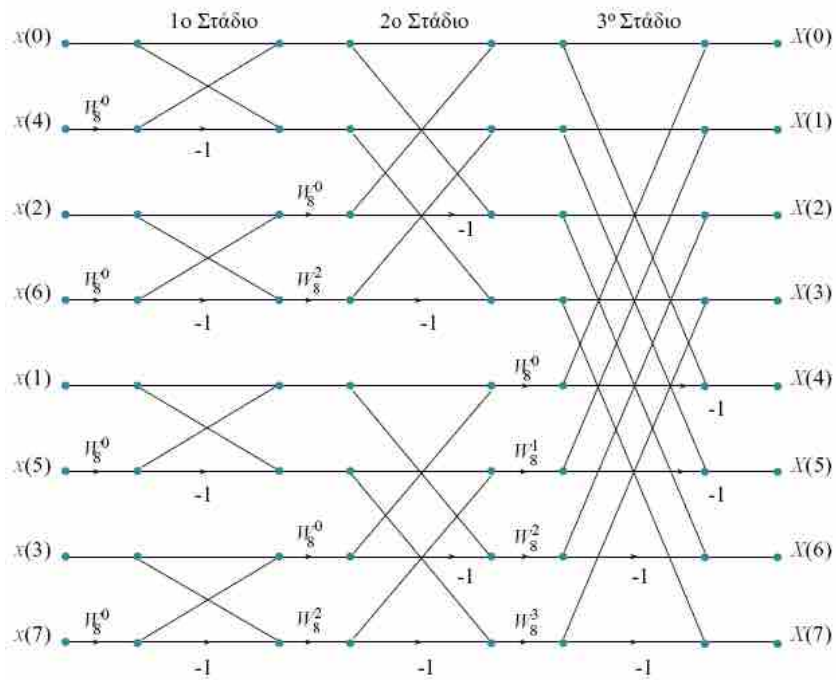
or 
$$X(k) = X_1(k) + W_N^k X_2(k)$$

but 
$$W_N^{k+\frac{N}{2}} = -W_N^k \quad W_N^{N\left(k+\frac{N}{2}\right)} = W_N^{nk} \quad X\left(k+\frac{N}{2}\right) = X_1(k) - W_N^k X_2(k)$$

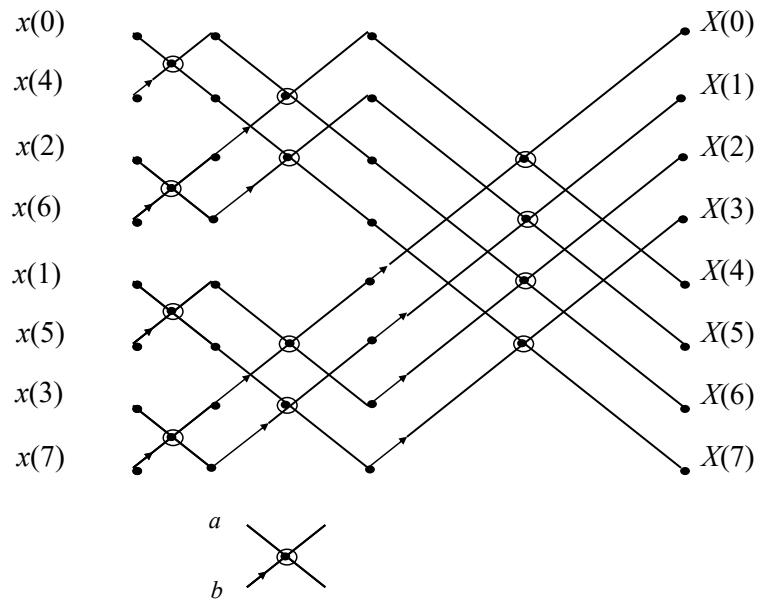




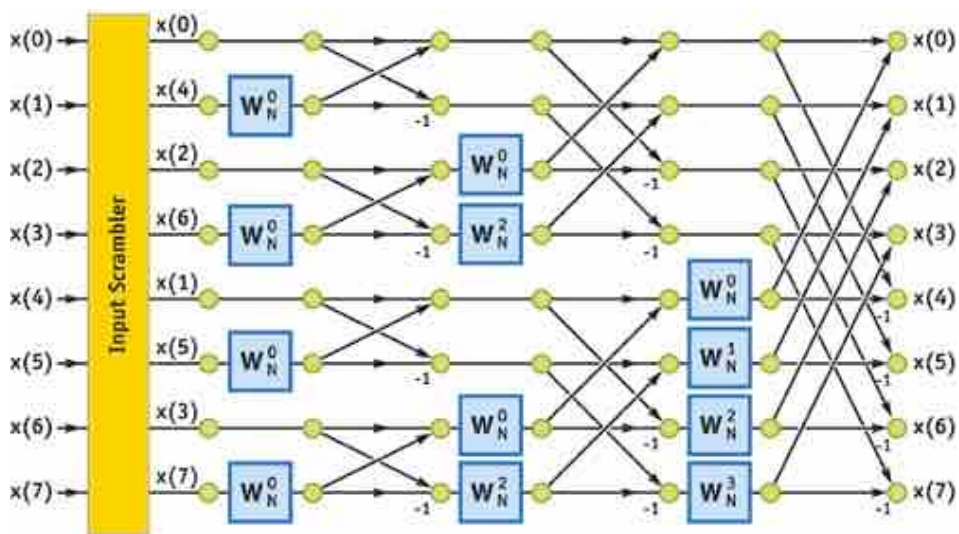
### 8-point FFT



## 8-point FFT (an alternative butterfly notation)



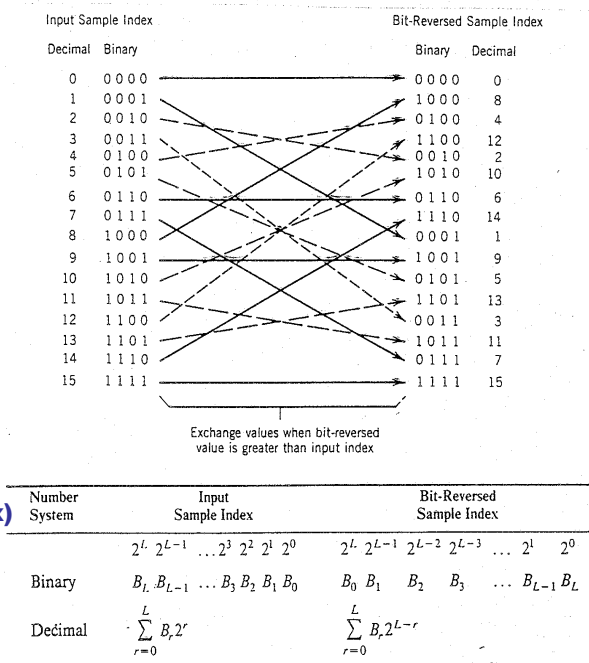
## 8-point DIT FFT flowgraph



**In-place computation means storing the results back in the original data locations.**

**In-place computation requires either input or output data shuffling.**

**Shuffling / Scrambling / Bit-reversal (for radix-2) / Digit-reversal (for any radix) are all synonymous.**



### Bit-Reversal

Κανονική διάταξη		Διάταξη από αντιστροφή της σειράς των bits	
Δεκαδικός	Δυαδικός	Δυαδικός	Δεκαδικός
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



## Computational Complexity (**complex** multiplications)

N	DFT	FFT	Λόγος Μιγαδικών Πολλαπλασιασμών (#DFT / #FFT)
	Πλήθος Μιγαδικών Πολλαπλασιασμών	Πλήθος Μιγαδικών Πολλαπλασιασμών	
2	4	1	4
4	16	4	4
8	64	12	5,3
16	256	32	8,0
32	1 024	80	12,8
64	4 096	192	21,3
128	16 384	448	36,6
256	65 536	1 024	64,0
512	262 144	2 304	113,8
1024	1 048 576	5 120	204,8
2048	4 194 304	11 264	372,4
4096	16 777 216	24 576	682,7
8192	67 108 864	53 248	1 260,3

### THE FAST FOURIER TRANSFORM (FFT) VS. THE DISCRETE FOURIER TRANSFORM (DFT)

■ The FFT is Simply an Algorithm for Efficiently Calculating the DFT

■ Computational Efficiency of an N-Point FFT:

◆ DFT:  $N^2$  Complex Multiplications

◆ FFT:  $(N/2) \log_2(N)$  Complex Multiplications

N	DFT Multiplications	FFT Multiplications	FFT Efficiency
256	65,536	1,024	64 : 1
512	262,144	2,304	114 : 1
1,024	1,048,576	5,120	205 : 1
2,048	4,194,304	11,264	372 : 1
4,096	16,777,216	24,576	683 : 1

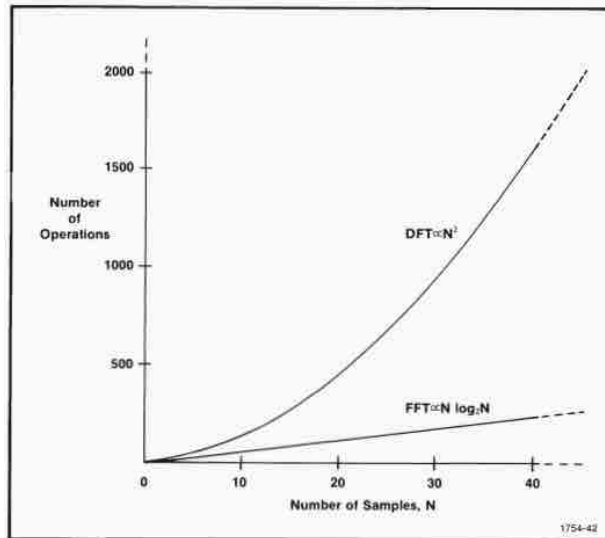
$10^{18}$  r  $\sim$  31.2 years  
 $30 \times 10^9$  n s  $\sim$  30 sec

$N$  |  $1000$  |  $10^6$  |  $10^9$   
 $N^2$  |  $10^6$  |  $10^{12}$  |  $10^{18}$   
 $N \log_2 N$  |  $10^4$  |  $20 \times 10^6$  |  $30 \times 10^9$

## FFT vs DFT

Number of complex multiplications

**FFT is simply an efficient algorithm for computing the DFT !**



## FFT Decimation In Frequency (DIF) Algorithm

FFT Decimation-In-Frequency (DIF) Algorithm  
(Sande-Tukey Algorithm)

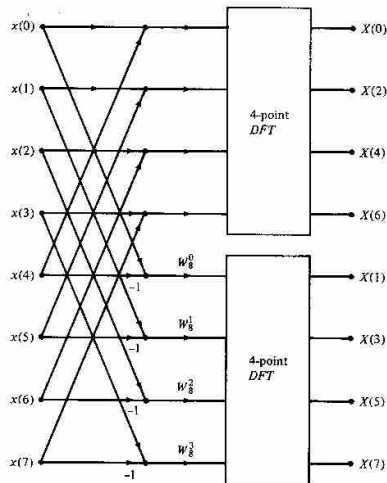
$$\begin{aligned}
 X(k) &= \sum_{n=0}^{N-1} x(n) W_N^{nk} = \sum_{n=0}^{\frac{N}{2}-1} x(n) W_N^{nk} + \sum_{n=\frac{N}{2}}^{N-1} x(n) W_N^{nk} \\
 &= \sum_{n=0}^{\frac{N}{2}-1} x(n) W_N^{nk} + \sum_{n=0}^{\frac{N}{2}-1} x(n + \frac{N}{2}) W_N^{(n+\frac{N}{2})k} \\
 &= \sum_{n=0}^{\frac{N}{2}-1} [x(n) + W_N^{(N/2)k} x(n + N/2)] W_N^{nk}
 \end{aligned}$$

By separating the frequency terms  $X(k)$  into even and odd samples of  $k$ , we get

$$\begin{aligned}
 X(6k) &= \sum_{n=0}^{\frac{N}{2}-1} [x(n) + x(n + \frac{N}{2})] W_{N/2}^{nk} \\
 X(2k+1) &= \sum_{n=0}^{\frac{N}{2}-1} [x(n) - x(n + \frac{N}{2})] W_N^n W_{N/2}^{nk}
 \end{aligned}$$

# FFT Decimation In Frequency (DIF) Algorithm

FFT Decimation-In-Frequency (DIF) Algorithm  
[Sande-Tukey Algorithm]

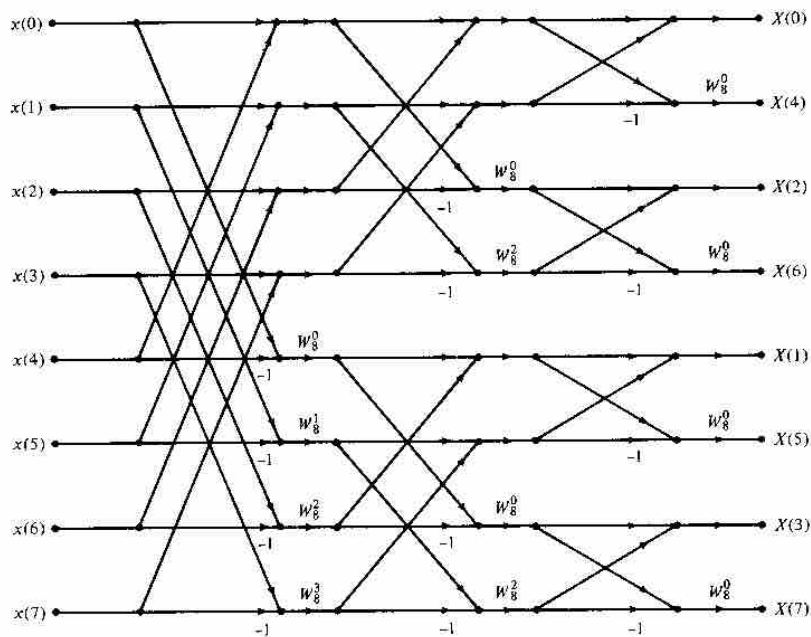


$$\begin{aligned}
 X(k) &= \sum_{n=0}^{N-1} x(n) W_N^{nk} = \sum_{n=0}^{N/2-1} x(n) W_N^{nk} + \sum_{n=N/2}^{N-1} x(n) W_N^{nk} \\
 &= \sum_{n=0}^{N/2-1} x(n) W_N^{nk} + \sum_{n=0}^{N/2-1} x(n + \frac{N}{2}) W_N^{(n + \frac{N}{2})k} \\
 &= \sum_{n=0}^{N/2-1} [x(n) + W_N^{(N/2)k} x(n + N/2)] W_N^{nk}
 \end{aligned}$$

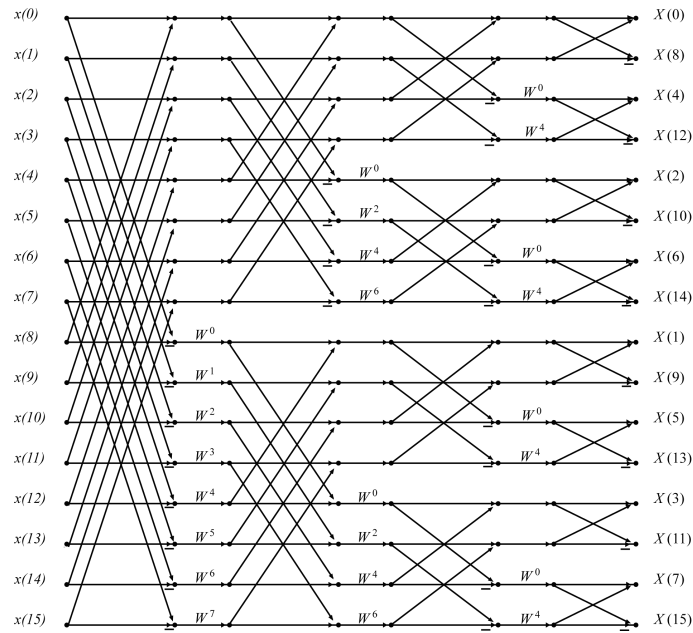
By separating the frequency terms  $X(k)$  into even and odd samples of  $k$ , we get

$$\begin{aligned}
 X(6k) &= \sum_{n=0}^{N/2-1} [x(n) + x(n + \frac{N}{2})] W_{N/2}^{nk} \\
 X(2k+1) &= \sum_{n=0}^{N/2-1} [x(n) - x(n + \frac{N}{2})] W_N^n W_{N/2}^{nk}
 \end{aligned}$$

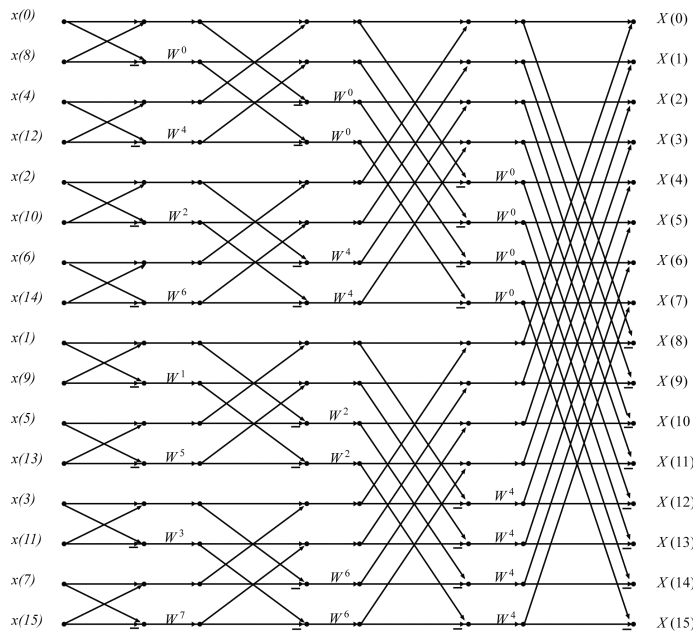
## 8-point DIF FFT flowgraph



## 16-point DIF FFT flowgraph



## 16-point DIT FFT flowgraph



### ΑΣΚΗΣΗ

Να υπολογιστεί το κόστος των πραγματικών πολ/τών και προσθίσεων για την αρίθμηση του DIT ή DIF FFT.

ΛΥΣΗ Έστω ότι το σήμα ανατελείται από  $N$  δείγματα.

Αναμφότερα στην περίπτωση του radix-2 FFT, παρατηρούμε ότι έχουμε  $\log_2 N$  στάδια και  $\frac{N}{2}$  πεταλούδες ανά στάδιο. Σε κάθε πεταλούδα έχουμε 1 μιγαδικό πολ/τό και 2 μιγαδικές προσθίσεις. Κάθε μιγαδικός πολ/τός απαιτεί 4 πραγματικούς πολ/τούς και 2 πραγματικές προσθίσεις, ενώ κάθε μιγαδική πρόσθεση απαιτεί 2 πραγμ. προσθίσεις. Επομένως έχουμε:

$$O_M = \frac{N}{2} \log_2 N \text{ μιγ. πολ/τοι} \rightarrow O_M = \left(\frac{N}{2} \log_2 N\right) \cdot 4 \text{ πραγμ. πολ/τοι} \rightarrow O_M = 2N \log_2 N \text{ Πραγματικοί προσθίσεις}$$

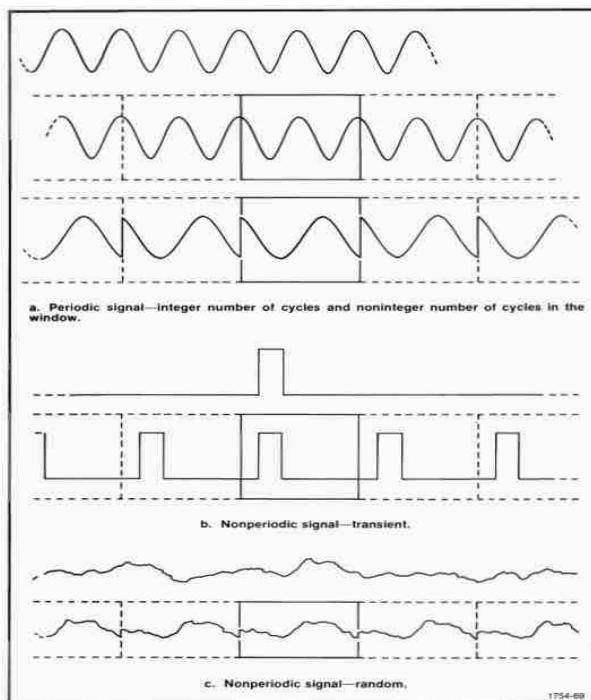
$$O_A = \underbrace{N \log_2 N}_{\substack{\text{άθρο} \\ \text{προσθίσεων} \\ \text{σε κάθε} \\ \text{πεταλούδα}}} + \underbrace{\frac{N}{2} \log_2 N}_{\substack{\text{άθρο} \\ \text{πολ/των} \\ \text{σε κάθε} \\ \text{πεταλούδα}}} \text{ μιγ. προσθίσεις} \rightarrow O_A = 2 \cdot (N \log_2 N) + 2 \left(\frac{N}{2} \log_2 N\right) = 3N \log_2 N \text{ πραγματικές προσθίσεις συνολικά}$$



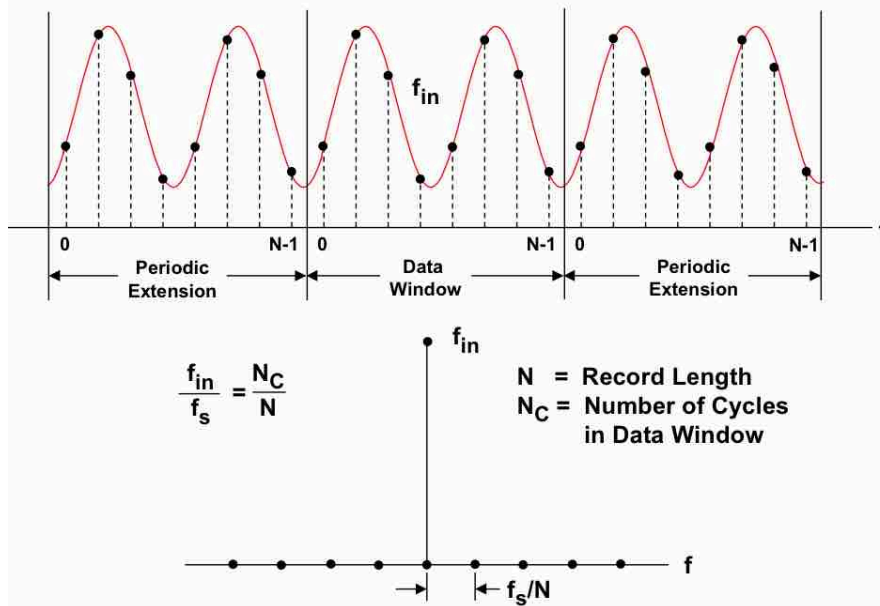
<https://www.youtube.com/watch?v=z7X6jgFnB6Y>

# WINDOWING

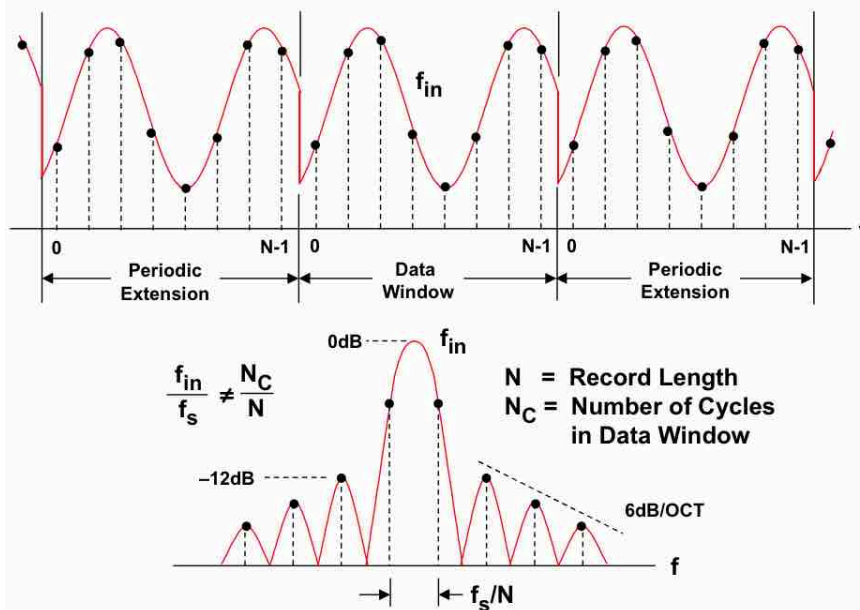
**The FFT assumes periodicity of the window**



## FFT OF SINEWAVE HAVING INTEGRAL NUMBER OF CYCLES IN DATA WINDOW

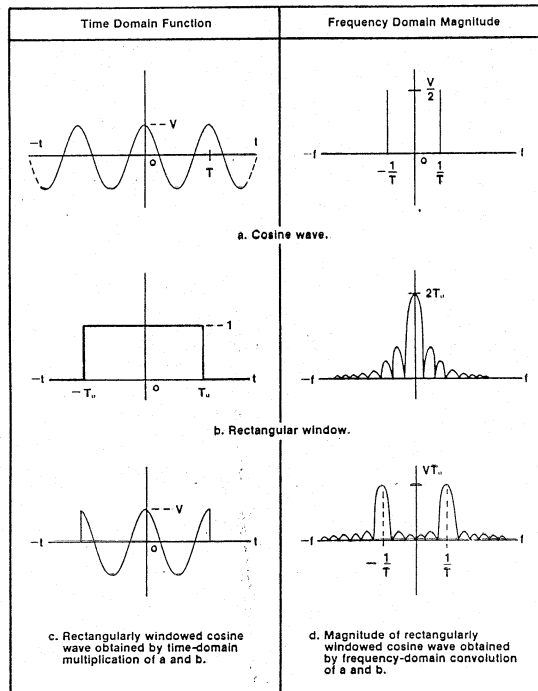


## FFT OF SINEWAVE HAVING NON-INTEGRAL NUMBER OF CYCLES IN DATA WINDOW

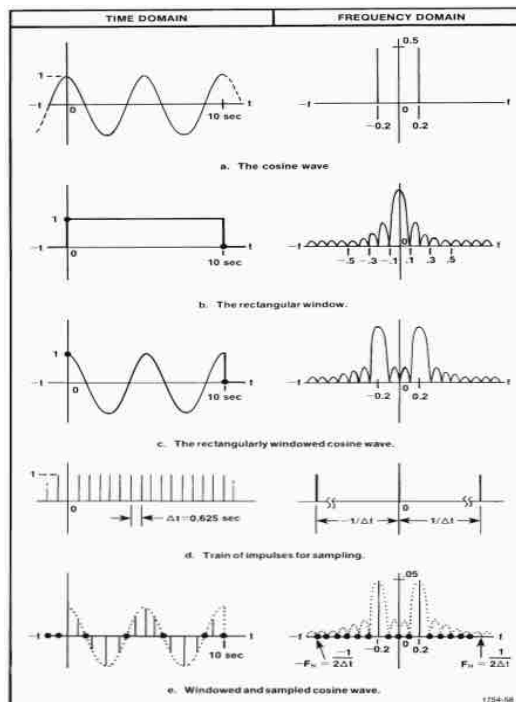


# Windows or Weighting Functions

## Rectangular window



Process of transforming a cosine wave having an **integer number of cycles in the window**



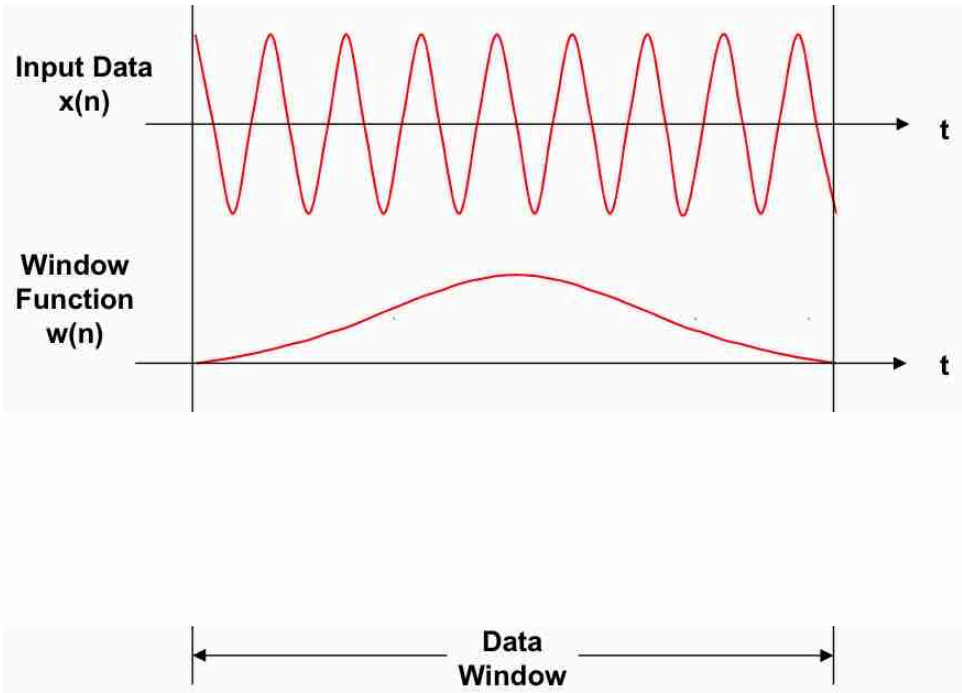
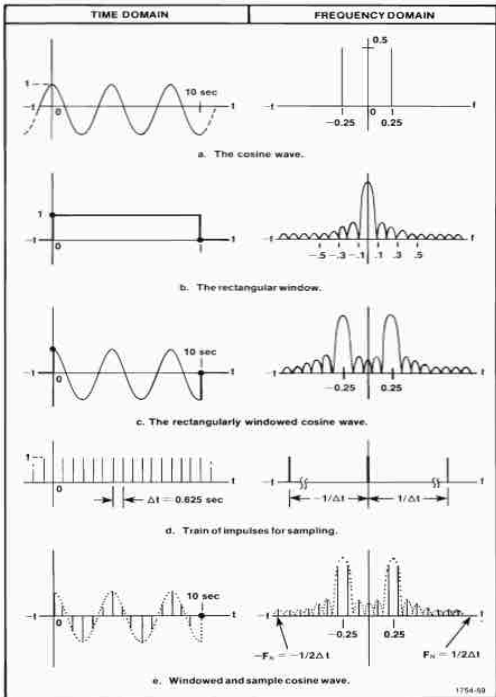


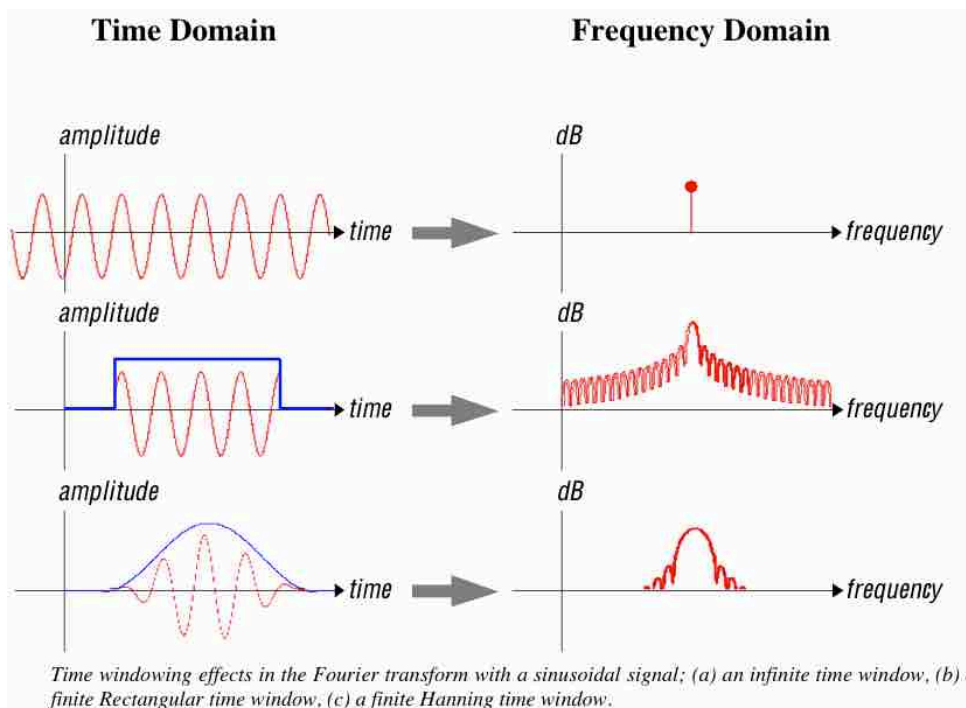
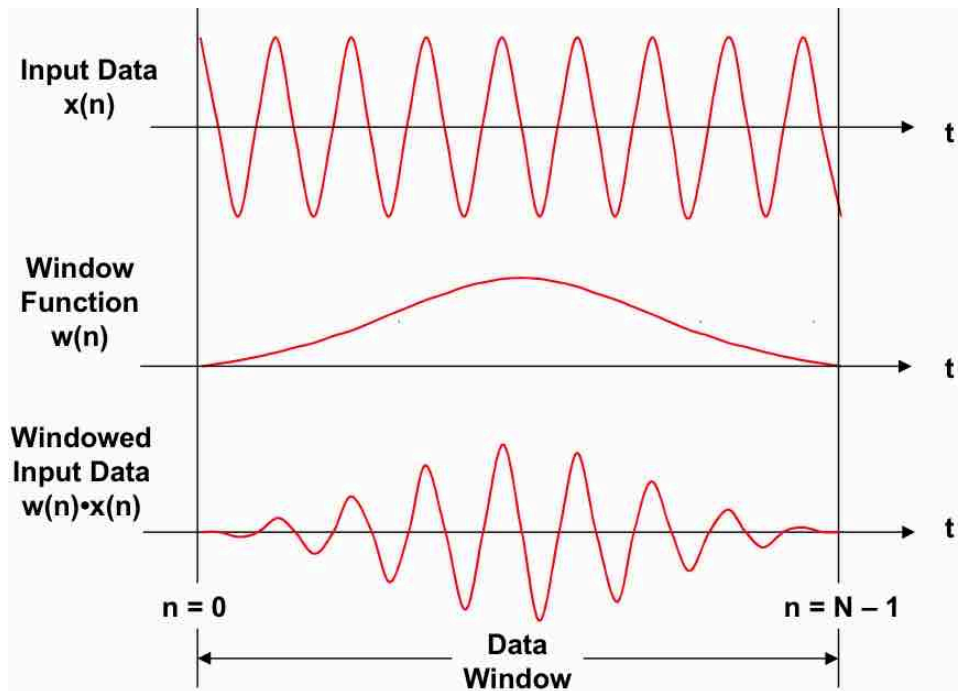
**Windowing can cause leakage !**

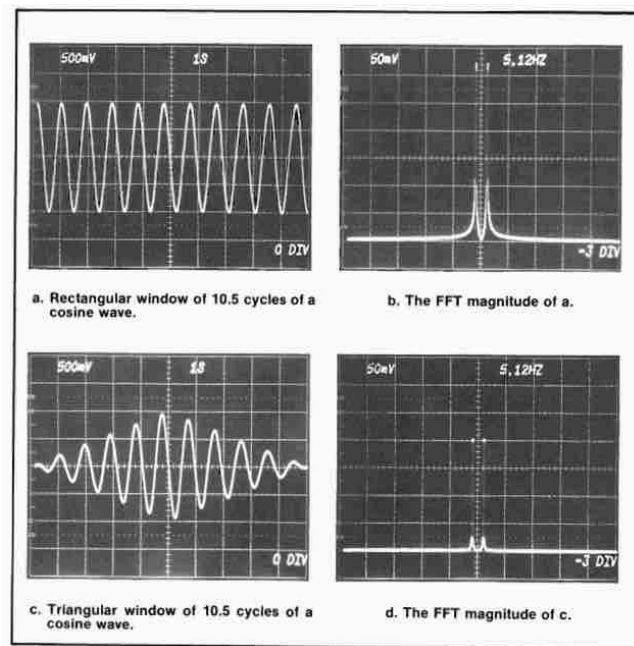
Process of transforming a cosine wave having a **non-integer** number of cycles in the window



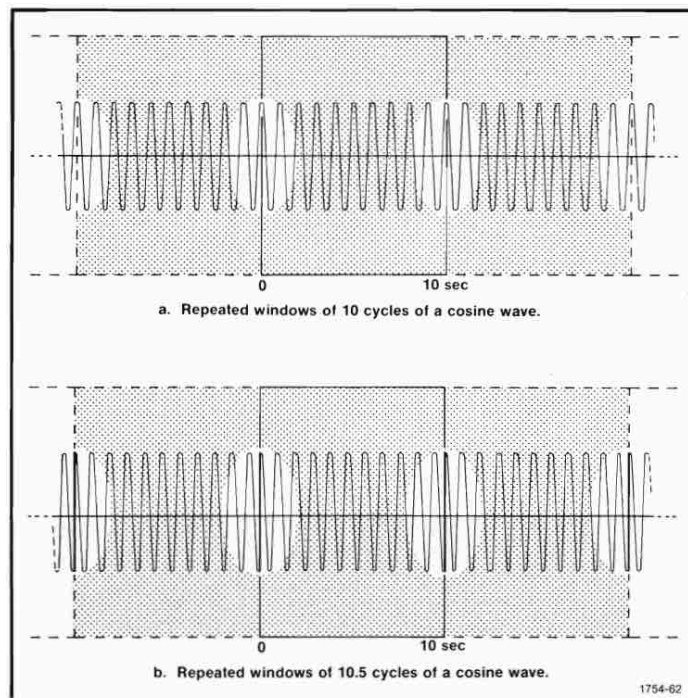
The **leakage error** comes about because the periods of the cosine wave are not harmonically related to window length, namely, there is a **non-integer** number of cycles in the window.



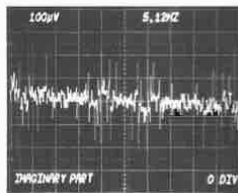
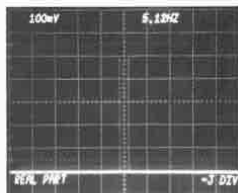
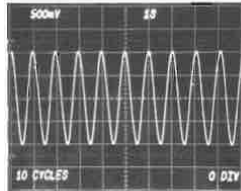




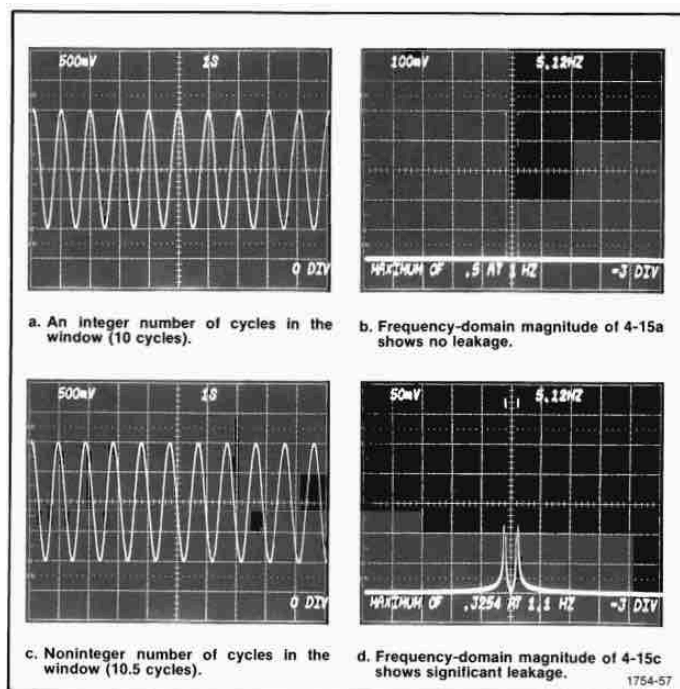
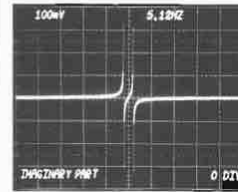
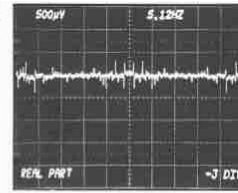
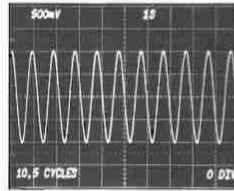
Window shape determines the degree of widening (main lobe) and rate of decay (side lobe size) of a spectral component.



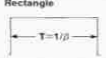
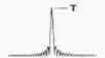










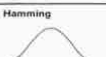

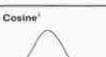



a. 10 cycles of a cosine wave appear as an even function—FFT frequency domain has a nonzero real part and an effectively zero imaginary part.



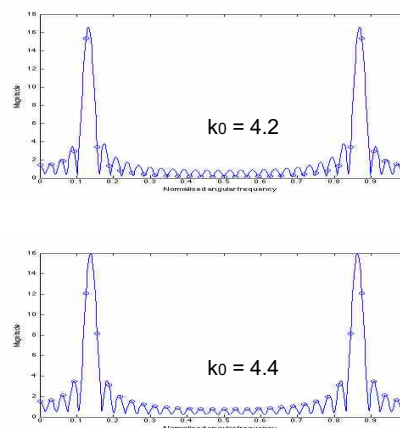
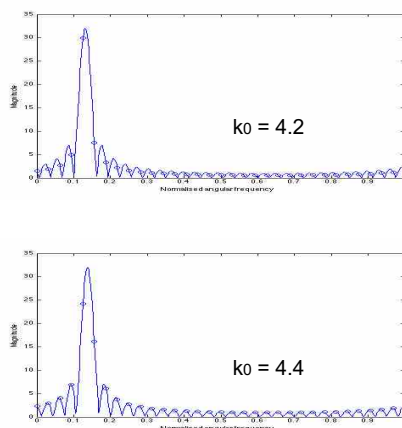
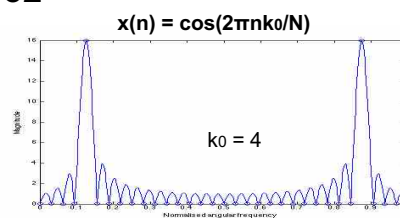
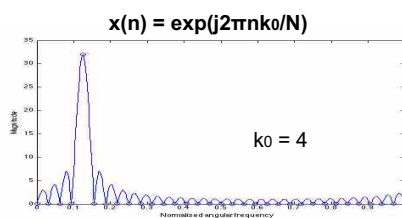
b. 10.5 cycles of a cosine wave appear as an odd function—FFT frequency domain has an effectively zero real part and a nonzero imaginary part.

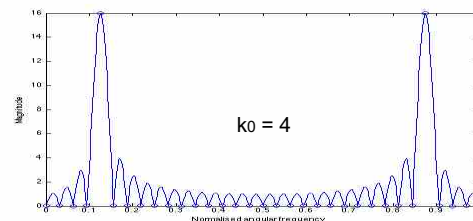
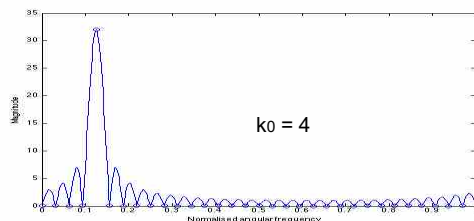
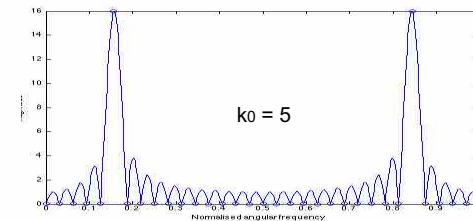
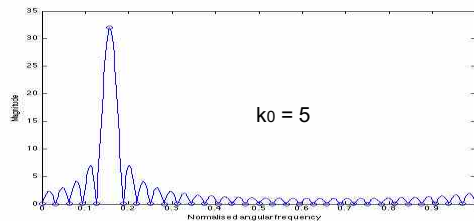
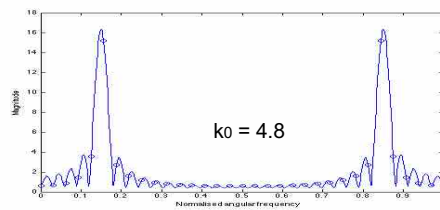
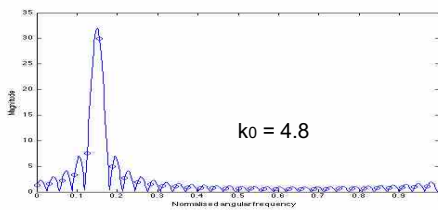
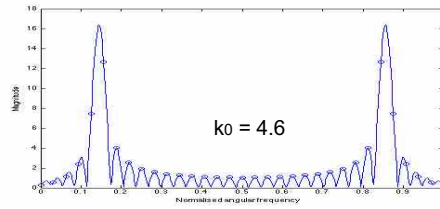
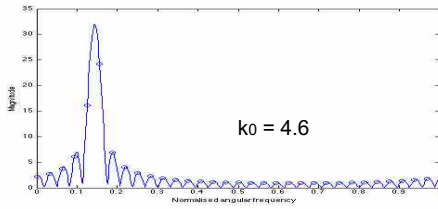
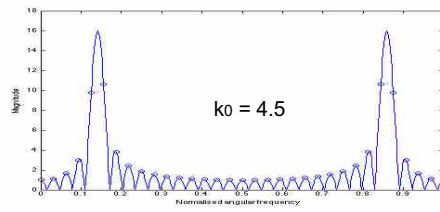
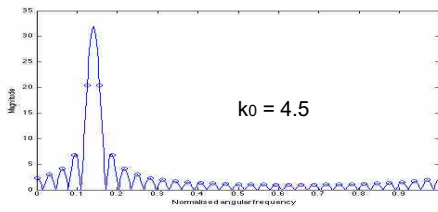


## Common data windows and their frequency-domain parameters

Unity Amplitude Window	Shape Equation	Frequency Domain Magnitude	Major Lobe Height	Highest Side Lobe (dB)	Bandwidth (3 dB)	Theoretical Roll-Off (dB/Octave)
 $T=1/\beta$	$A=1$ for $t=0$ to $T$		$T$	-13.2	$0.86\beta$	6
	$A=0.5(1-\cos 2\pi t/T)$ for $t=0$ to $T/10$ and $t=9T/10$ to $T$ $A=1$ for $t=T/10$ to $9T/10$		$0.9 T$	-13.5	$0.95\beta$	18 (beyond $5\beta$ )
	$A=\sin 2\pi 0.5t/T$ for $t=0$ to $T$		$0.64 T$	-22.4	$1.15\beta$	12
	$A=2t/T$ for $t=0$ to $T/2$ $A=-2t/T + 2$ for $t=T/2$ to $T$		$0.5 T$	-26.7	$1.27\beta$	12
	$A=0.5(1-\cos 2\pi t/T)$ for $t=0$ to $T$		$0.5 T$	-31.6	$1.39\beta$	18
	$A=\sin^2 2\pi 0.5t/T$ for $t=0$ to $T$		$0.42 T$	-39.5	$1.61\beta$	24
	$A=0.08 + 0.46(1-\cos 2\pi t/T)$ for $t=0$ to $T$		$0.54 T$	-41.9	$1.26\beta$	6 (Beyond $5\beta$ )
	$A=(0.5(1-\cos 2\pi t/T))^2$ for $t=0$ to $T$		$0.36 T$	-46.9	$1.79\beta$	30
	$A=1-6(2t/T-1)^2+6(2t/T-1)^3$ for $t=T/4$ to $3T/4$ $A=2(1- 2t/T-1 )^3$ for $t=0$ to $T/4$ and $t=3T/4$ to $T$		$0.37 T$	-53.2	$1.81\beta$	24

**N=32**





**Matlab code**

```
N = input('Number of points = ');
k0 = input('Frequency = ');
n = 0:N-1;
x = exp(j*(2*pi)/N)*k0*n);
X = fft(x);
XE = fft(x, 512);
```

```
% Plot the magnitude of X
L = 0:511;
plot(L/512, abs(XE))
hold
plot(n/N, abs(X), 'o')
xlabel('Normalised angular frequency')
ylabel('Magnitude')
```

## REAL TIME FFT CONSIDERATIONS

- Signal Bandwidth
- Sampling Frequency,  $f_s$
- Number of Points in FFT,  $N$
- Frequency Resolution =  $f_s / N$
- Maximum Time to Calculate N-Point FFT =  $N / f_s$
- Fixed-Point vs. Floating Point DSP
- Radix-2 vs. Radix-4 Execution Time
- FFT Processing Gain =  $10 \log_{10}(N / 2)$
- Windowing Requirements



<https://www.youtube.com/watch?v=zKKGGA30bHG0>