PHOBIA GLOSSARY (FROM THE LOS ANGELES TIMES)

AUTHORITIES SAY A SEEMINGLY ENDLESS VARIETY OF OBJECTS OR PHENOMENON CAN TRIGGER PHOBIAS. MOST OF THEM – LIKE FEAR OF HEIGHTS, OR FEAR OF WATER – DO NOT INCAPACITATE AN INDIVIDUAL. OTHERS DO. SOME PHOBIAS ARE UNIQUE TO INDIVIDUALS; MOST OTHERS HAVE APPEARED FREQUENTLY ENOUGH TO HAVE BEEN GIVEN A NAME. BELOW IS A LIST OF MEDICALLY RECOGNIZED PHOBIAS, WITH THEIR LATIN OR GREEK NAMES PRESENTED FIRST, FOLLOWED BY THEIR ENGLISH TRANSLATION.

- **ARACHNEPHOBIA -** Fear of Spiders.
- **AMAXOPHOBIA or HAMAXOPHOBIA** – Fear of vehicles.
- **ANDROPHOBIA** – Fear of Males.
- **AUTOPHOBIA** – Fear of Self.
- **AUTOMYSOPHOBIA –** Fear of being dirty.
- **BELONEPHOBIA –** Fear of pins.
- **BROMIDOSIPHOBIA or**
- **OSPHRESIOPHOBIA or OSMOPHOBIA –** Fear of odors.
- **CYNOPHOBIA –** Fear of dogs.
- **ENTOMOPHOBIA –** Fear of Insects.
- **ERGASIOPHOBIA or PONOPHOBIA –** Fear of work.
- **GAMOPHOBIA –** Fear of marriage.
- **GEPHYROPHOBIA –** Fear of crossing a bridge.
- **GRAPHOPHOBIA –** Fear of writing.
- **GYNEPHOBIA –** Fear of women.
- **HELIOPHOBIA –** Fear of sun.
- **HODCPHOBIA –** Fear of traveling.
- **HYGROPHOBIA –** Fear of dampness.
- **HYPNOPHOBIA –** Fear of sleep.
- **ICHTHYOPHOBIA –** Fear of fish.
- **MYSOPHOBIA or RHYPOPHOBIA –** Fear of dirt.
- **OMBROPHOBIA –** Fear of rain.
- **PANPHOBIA –** Fear of everything.
- **PENIAPHOBIA –** Fear of poverty.
- **PHARMACOPHOBIA –** Fear of drugs.
- **PHOBOPHOBIA –** Fear of phobias.
- **SIDERODROMOPHONIA –** Fear of trains.
- **SPECTRAPHOBIA –** Fear of *Fourier Transforms*.
- **VACCINOPHOBIA –** Fear of vaccination.

$$\mathscr{F}\{g(t)\} = G(f) = \int_{-\infty}^{\infty} g(t)e^{-i2\pi ft}\,dt$$

$$\mathscr{F}^{-1}\{G(f)\} = g(t) = \int_{-\infty}^{\infty} G(f)e^{i2\pi ft}\,df$$

FAST FOURIER TRANSFORM ALGORITHMS

COOLEY-TUKEY, GOOD-THOMAS, WINOGRAD, BLUESTEIN, & RADER

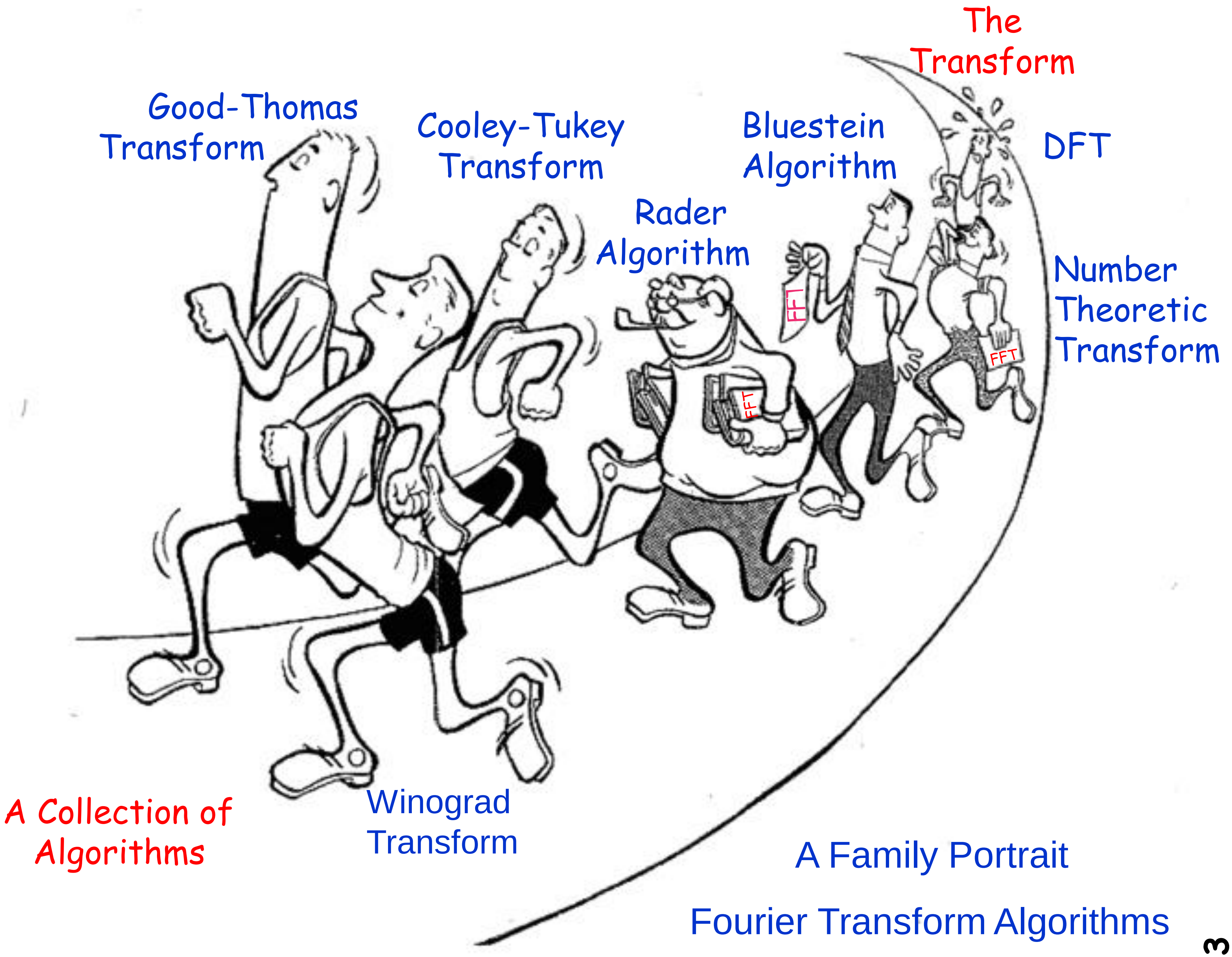
(AND A BIT OF SPECTRAL ANALYSIS WITH THE FFT)

fred harris

San Diego State University
College of Engineering

13 March 2014

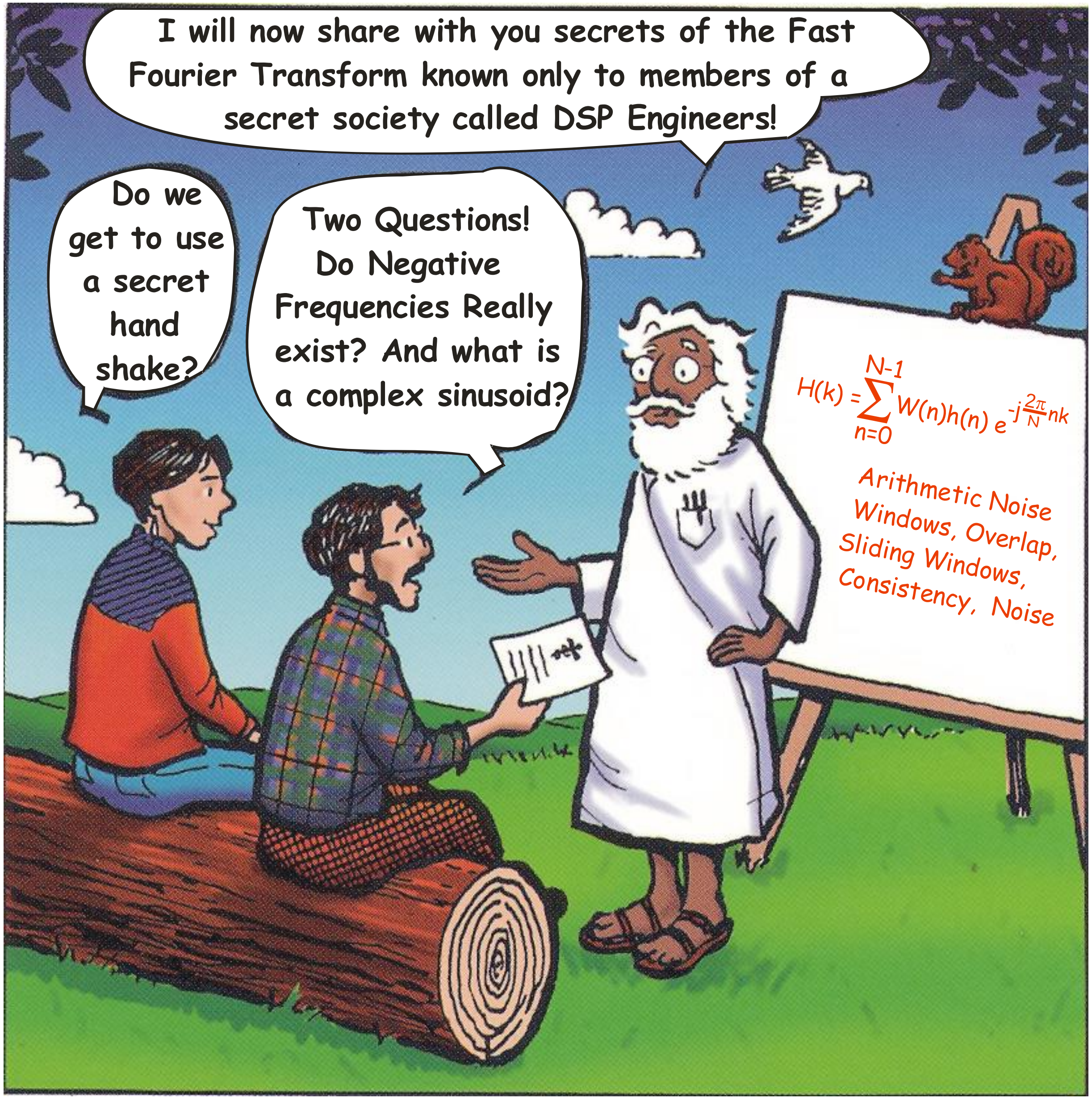
Wireless Innovation Forum



A Family Portrait

Fourier Transform Algorithms

INTERESTING FFT TRIVIA FROM THE FFT GURU



WHO LIVES IN FFT-LAND?

➢Cooley-Tukey Algorithm

Maps a 1-dimensional Transform of composite length N into coupled two dimensional Transforms of length $N_1 x N_2$

$N=N_1*N_2$, N_1 and N_2 Arbitrary

Some Engineers

➢Good-Thomas (Prime Factor) Algorithm

Maps a 1-dimensional Transform of composite length N into uncoupled two dimensional Transforms of length $N_1 x N_2$

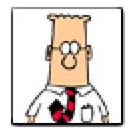
$N=N_1*N_2$, $GCD(N_1,N_2)=1$, (N_1,N_2 Relatively Prime)))

Some Engineers

➢Bluestein (Chirp) Algorithm

Maps a 1-dimensional Transform of length N into a Linear Convolution of Length 2N-1. Convolution Implemented in Transform of length L.

N Arbitrary, L>2N-1



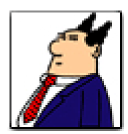
Some Engineers

➢Rader Algorithm

A Few
Super Heroes

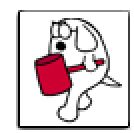
Maps a one dimensional transform of Prime Length N into a Circular Convolution of length N-1

N-Prime



A Pointed Hair Boss

A Pretty Savvy Dog



➢Winograd Algorithm

Maps short Rader Circular Convolution Transform, to Computationally Efficient Winograd Convolver, (Workload ~ 2N)

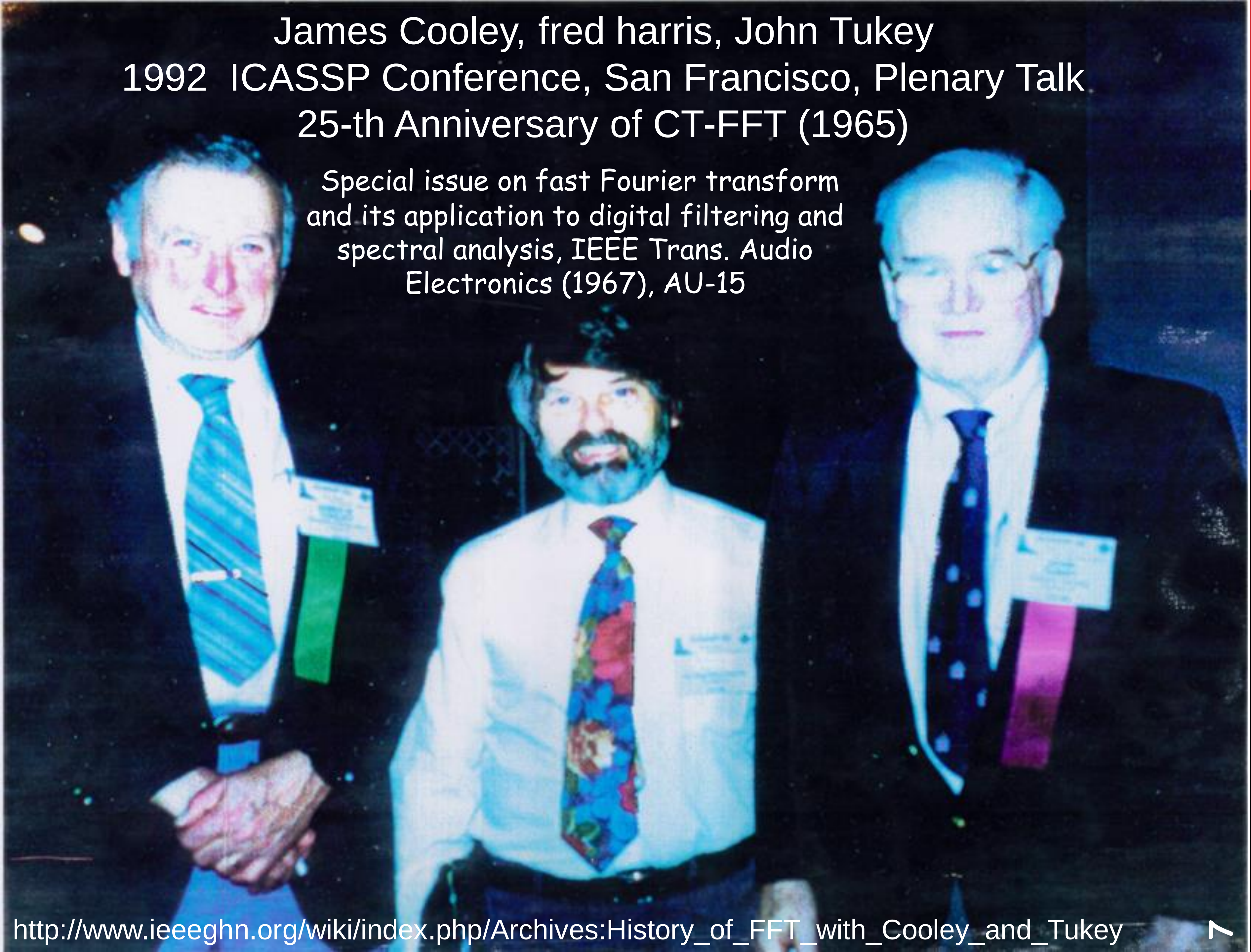
N-Small Prime or Power of Prime

A Very Smart
Trash Collector

➢Number Theoretic Transform

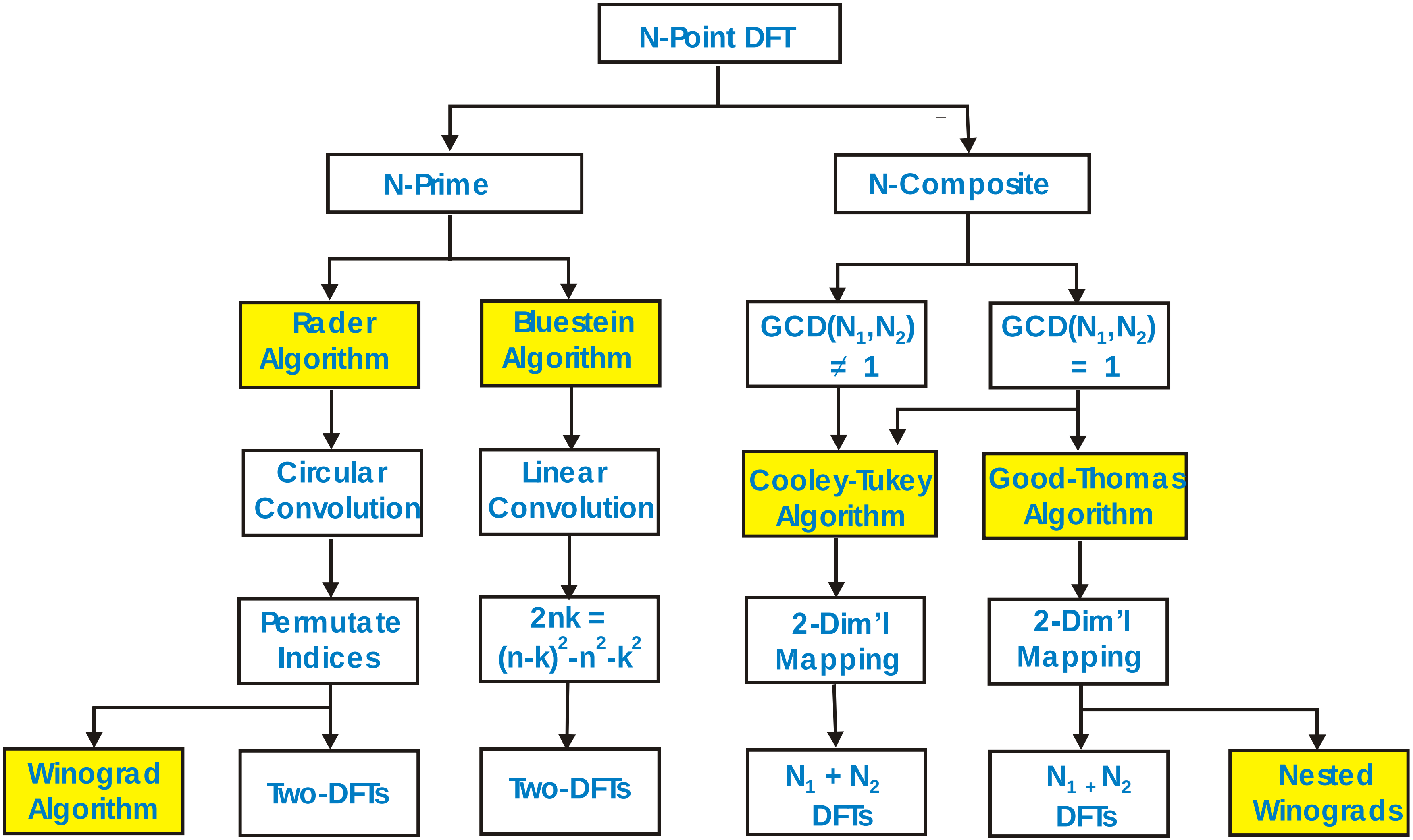
In Certain Galois Fields, the N-th Root of Unity is the integer 2
Transforms; implemented with integer arithmetic and no Multiplies!
Transform Supports Convolution

James Cooley, fred harris, John Tukey
1992 ICASSP Conference, San Francisco, Plenary Talk
25-th Anniversary of CT-FFT (1965)



Special issue on fast Fourier transform and its application to digital filtering and spectral analysis, IEEE Trans. Audio Electronics (1967), AU-15

http://www.ieeeghn.org/wiki/index.php/Archives:History_of_FFT_with_Cooley_and_Tukey


N-Point DFT
N-Prime
N-Composite
Rader Algorithm
Bluestein Algorithm
GCD(N1,N2) ≠ 1
GCD(N1,N2) = 1
Circular Convolution
Linear Convolution
Cooley-Tukey Algorithm
Good-Thomas Algorithm
Permutate Indices
2nk = (n-k)2-n2-k2
2-Dim'l Mapping
2-Dim'l Mapping
Winograd Algorithm
Two-DFTs
Two-DFTs
N1 + N2 DFTs
N1 + N2 DFTs
Nested Winograds

FFT-TREE

INVERSE FOURIER TRANSFORM

Inverse Fourier Transform

$$h(t) = \int_{\text{support}} H(\omega) e^{j\omega t} d\omega / 2\pi$$

Inverse Fourier Series

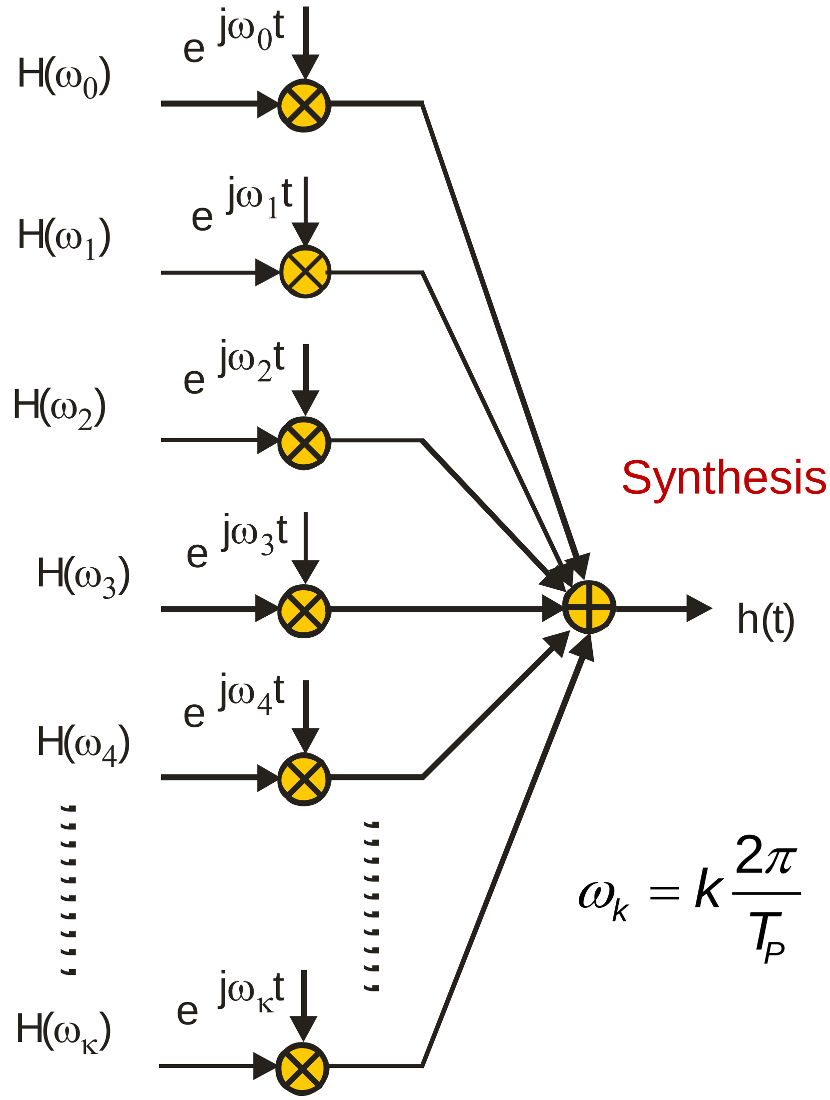
$$h(t) = \sum_k C_k e^{j\omega_k t}; \omega_k = k \frac{2\pi}{T_{Period}}$$

Inverse Discrete Time Series

$$h(n) = \frac{1}{2\pi} \int_0^{2\pi} H(\theta) e^{j\theta n} d\theta / 2\pi; \theta = 2\pi \frac{f}{f_S}$$

Inverse Discrete Fourier Transform

$$h(n) = \frac{1}{N} \sum_{k=0}^{N-1} H(k) e^{j\theta_k n}; \theta_k = \frac{2\pi}{N} k$$



$$\omega_k = k \frac{2\pi}{T_P}$$

THE FOURIER TRANSFORM

Forward Fourier Transform

$$H(\omega) = \int_{\text{support}} h(t) e^{-j\omega t} dt$$

$$\omega_k = k \frac{2\pi}{T_P}$$

Forward Fourier Series

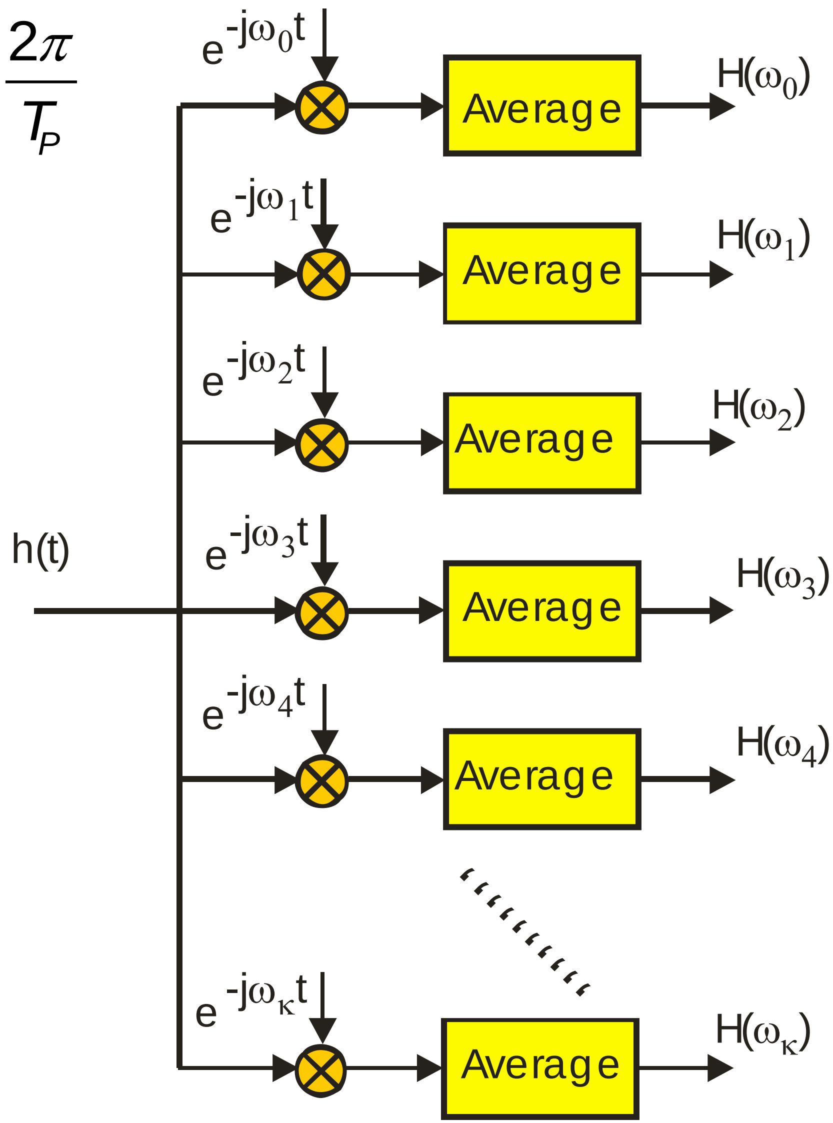
$$C_k = \frac{1}{T_{\text{Period}}} \int_{T_{\text{Period}}} h(t) e^{-j\omega_k t} dt; \quad \omega_k = k \frac{2\pi}{T_{\text{Period}}}$$

Forward Discrete Time Fourier Transform

$$H(\theta) = \sum_{\text{support}} h(n) e^{-j\theta n}; \quad \theta = 2\pi \frac{f}{f_S}$$

Forward Discrete Fourier Transform

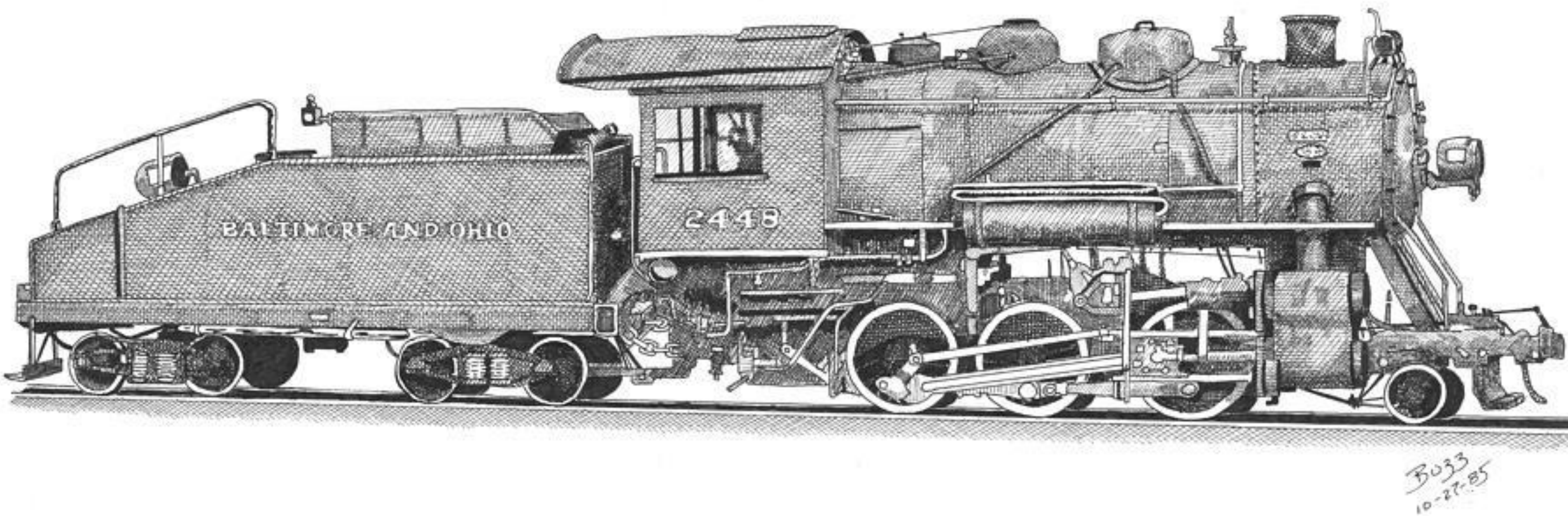
$$H(k) = \sum_{n=0}^{N-1} h(n) e^{-j\theta_k n}; \quad \theta_k = 2\pi \frac{f}{f_S}\bigg|_{f=k\frac{f_S}{N}} = k \frac{2\pi}{N}$$



Analysis

The word Orthogonal
Is used a lot around here!

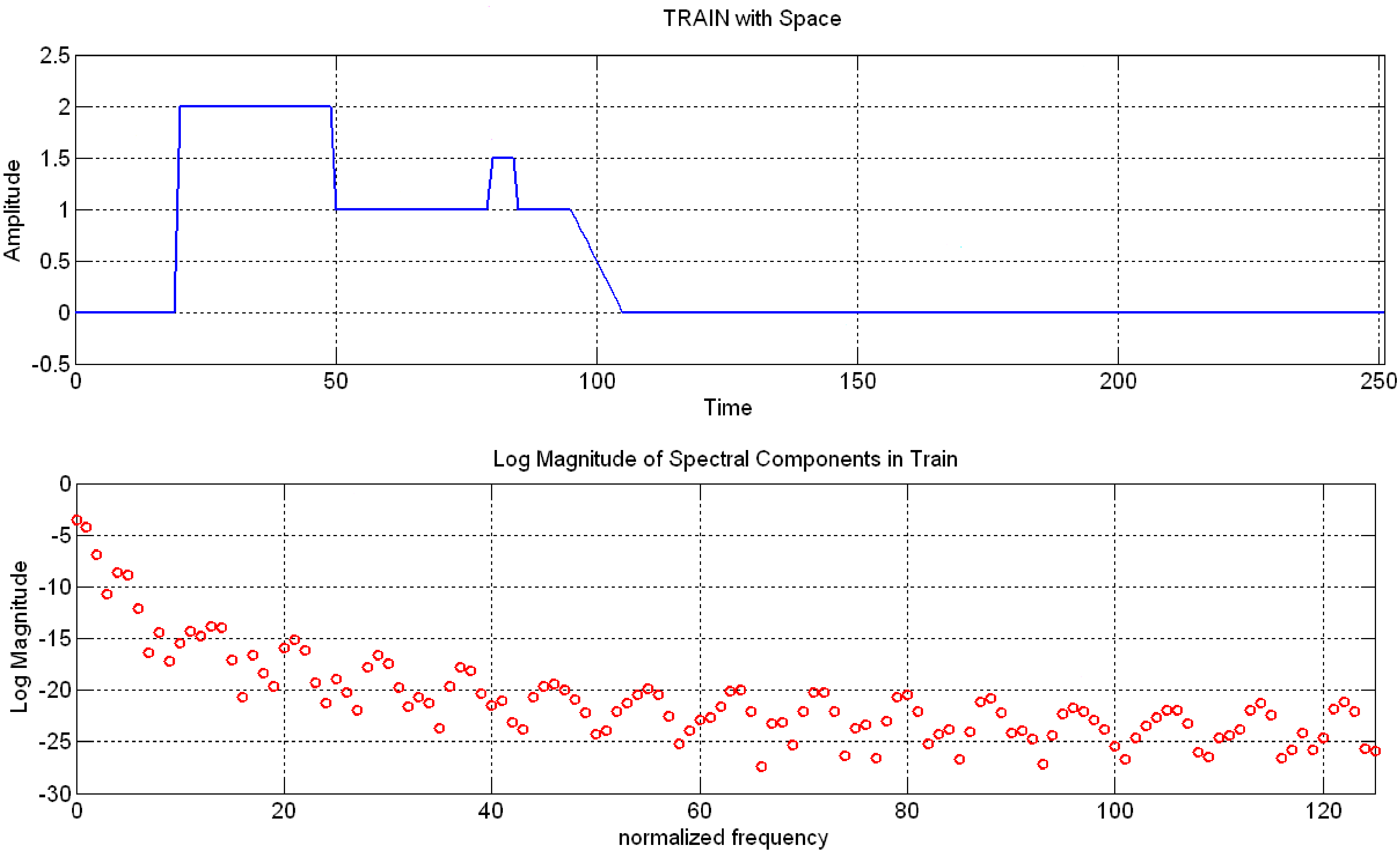
Steam Locomotive



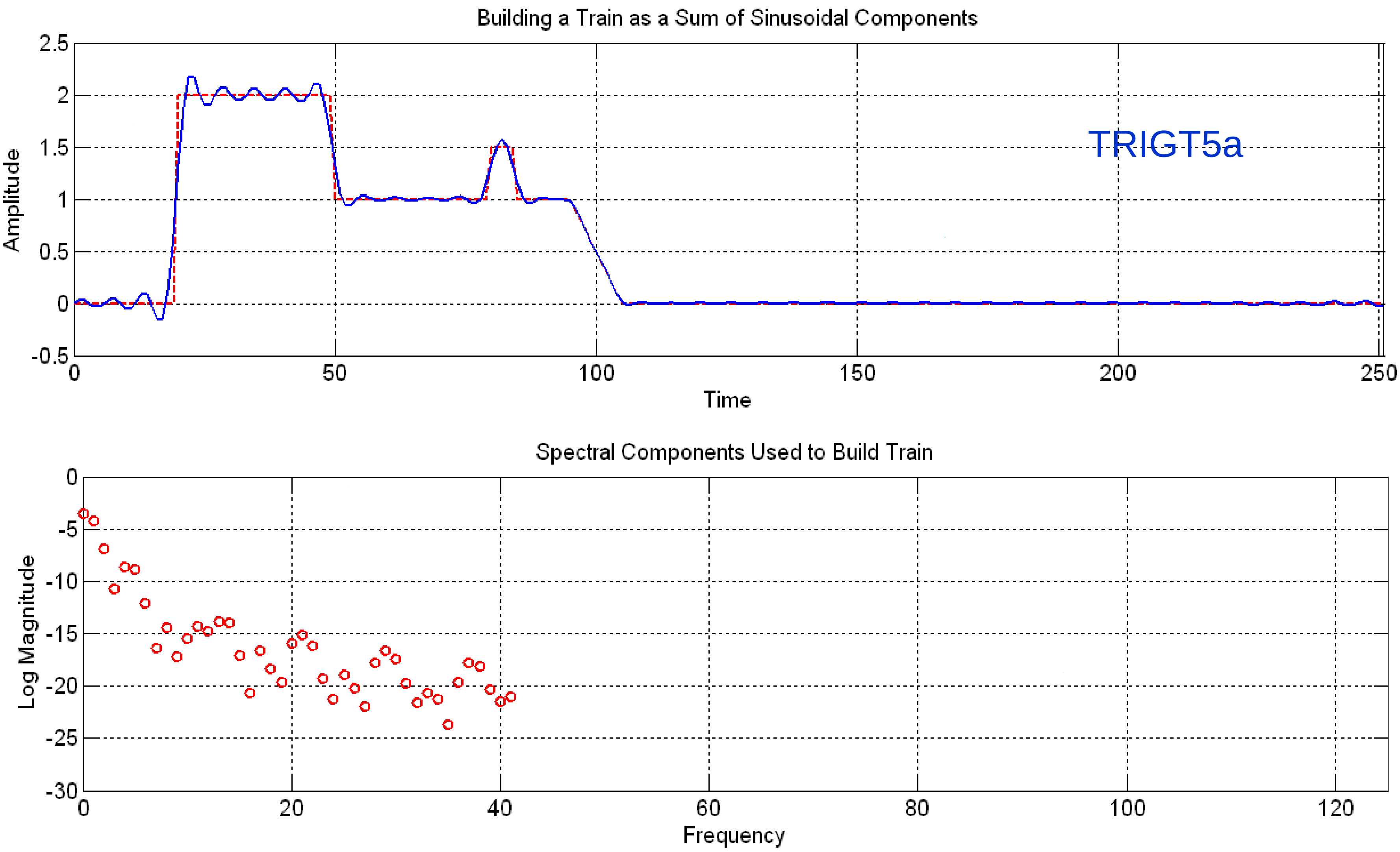
Also known as A Choo-Choo

PULSE TRAIN AND ITS FOURIER COEFFICIENTS

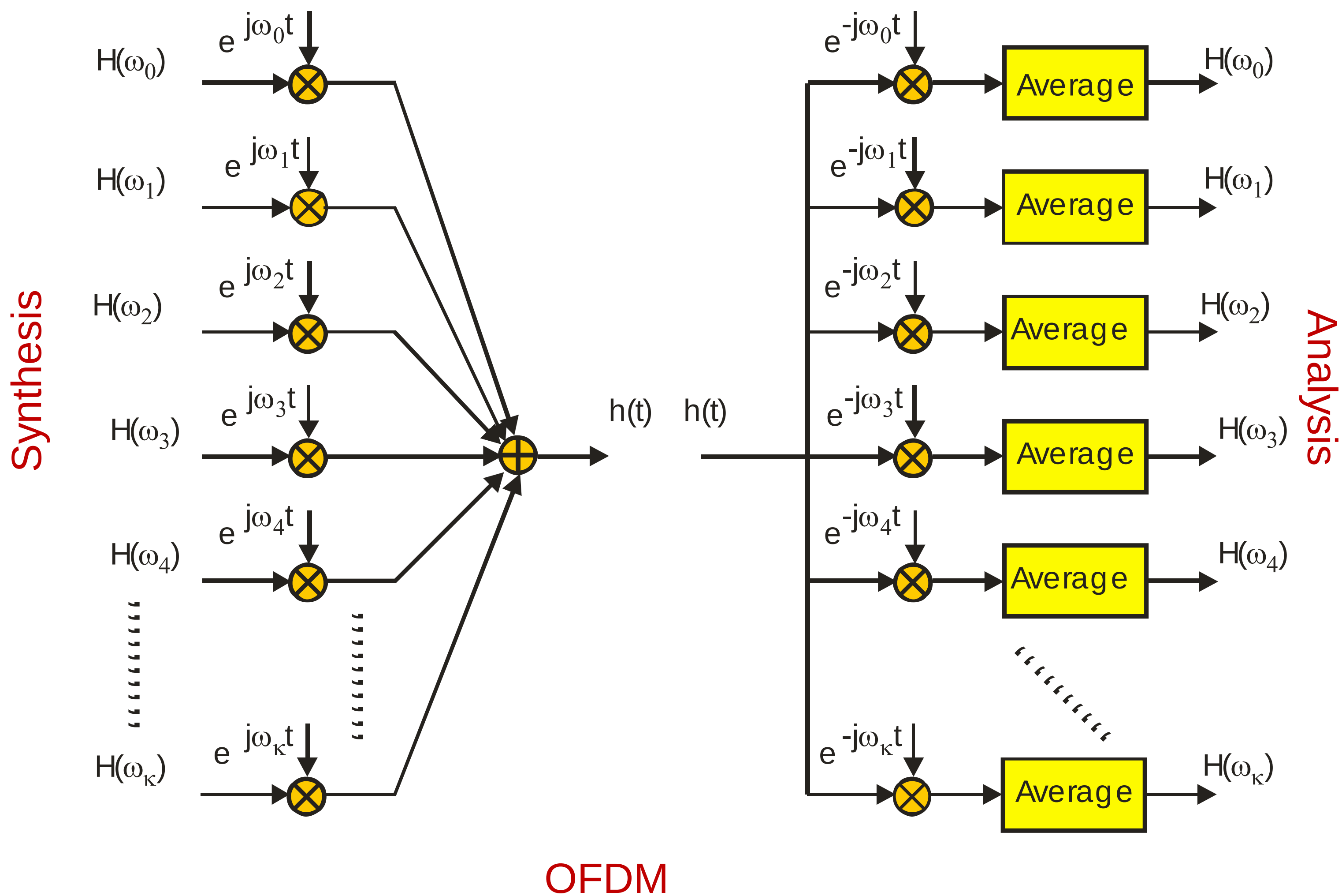
TRIGT5a



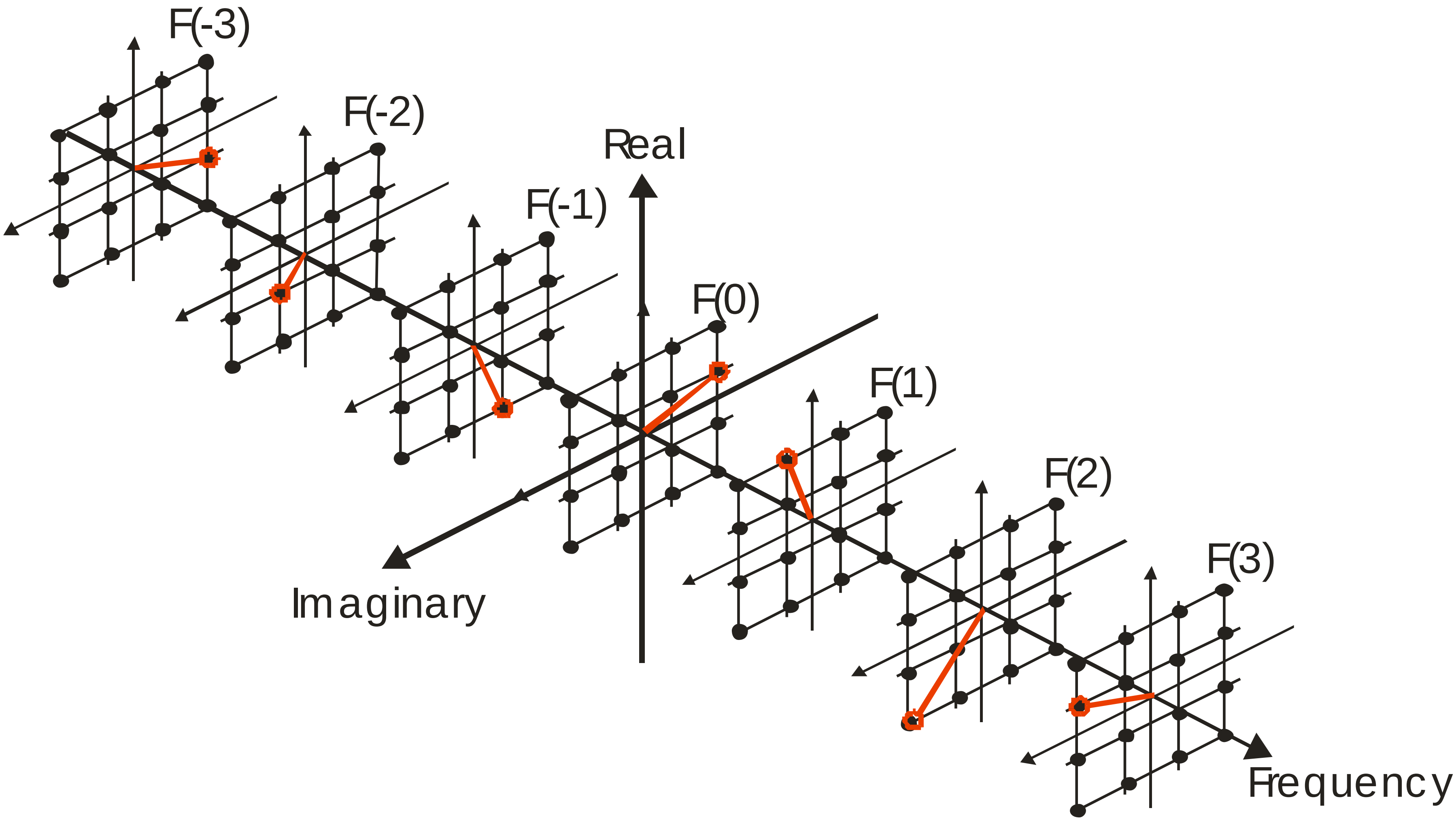
PULSE TRAIN AND ITS FIRST 41 FOURIER COEFFICIENTS



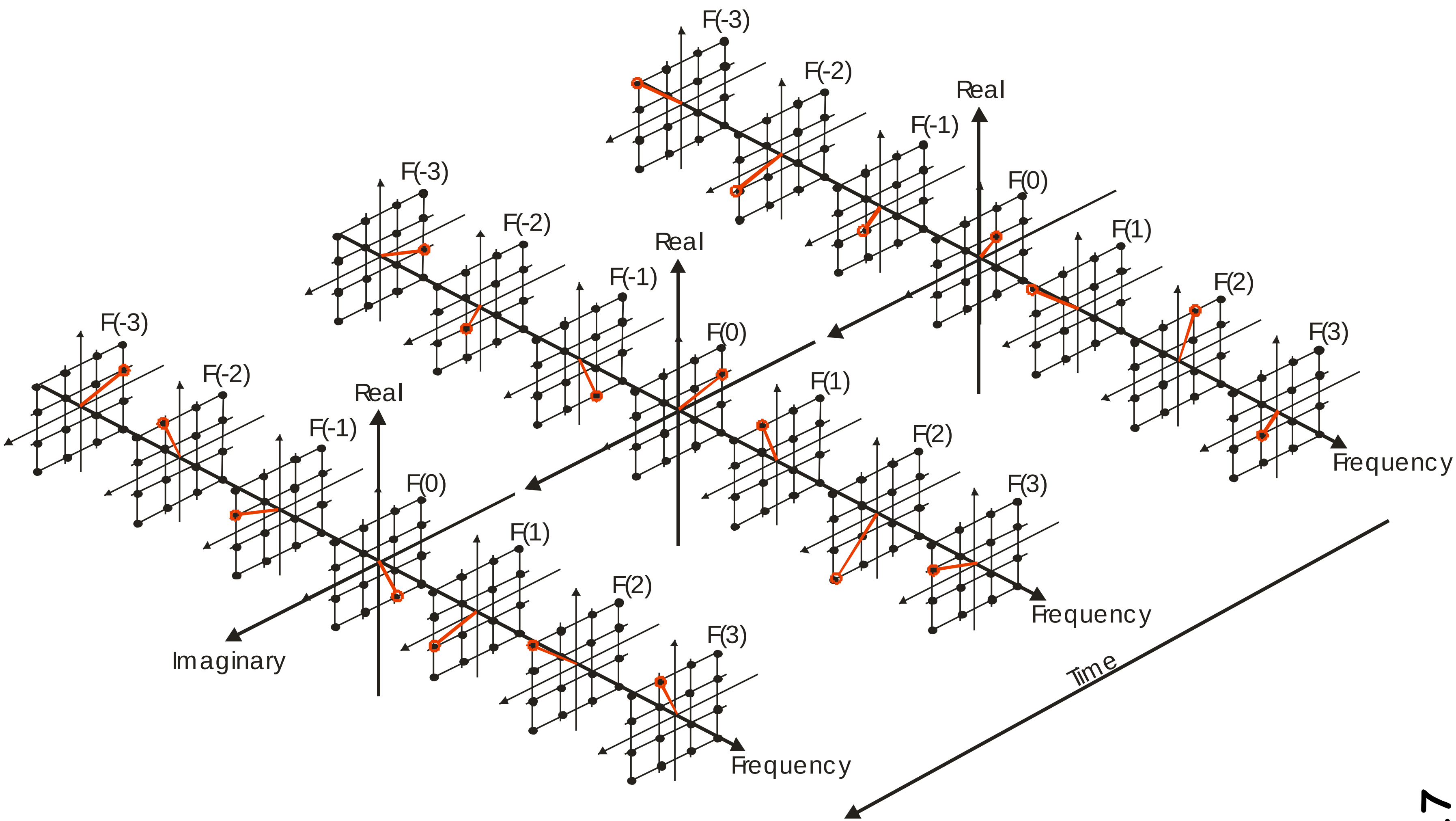
CASCADE INVERSE AND FORWARD FOURIER TRANSFORMS



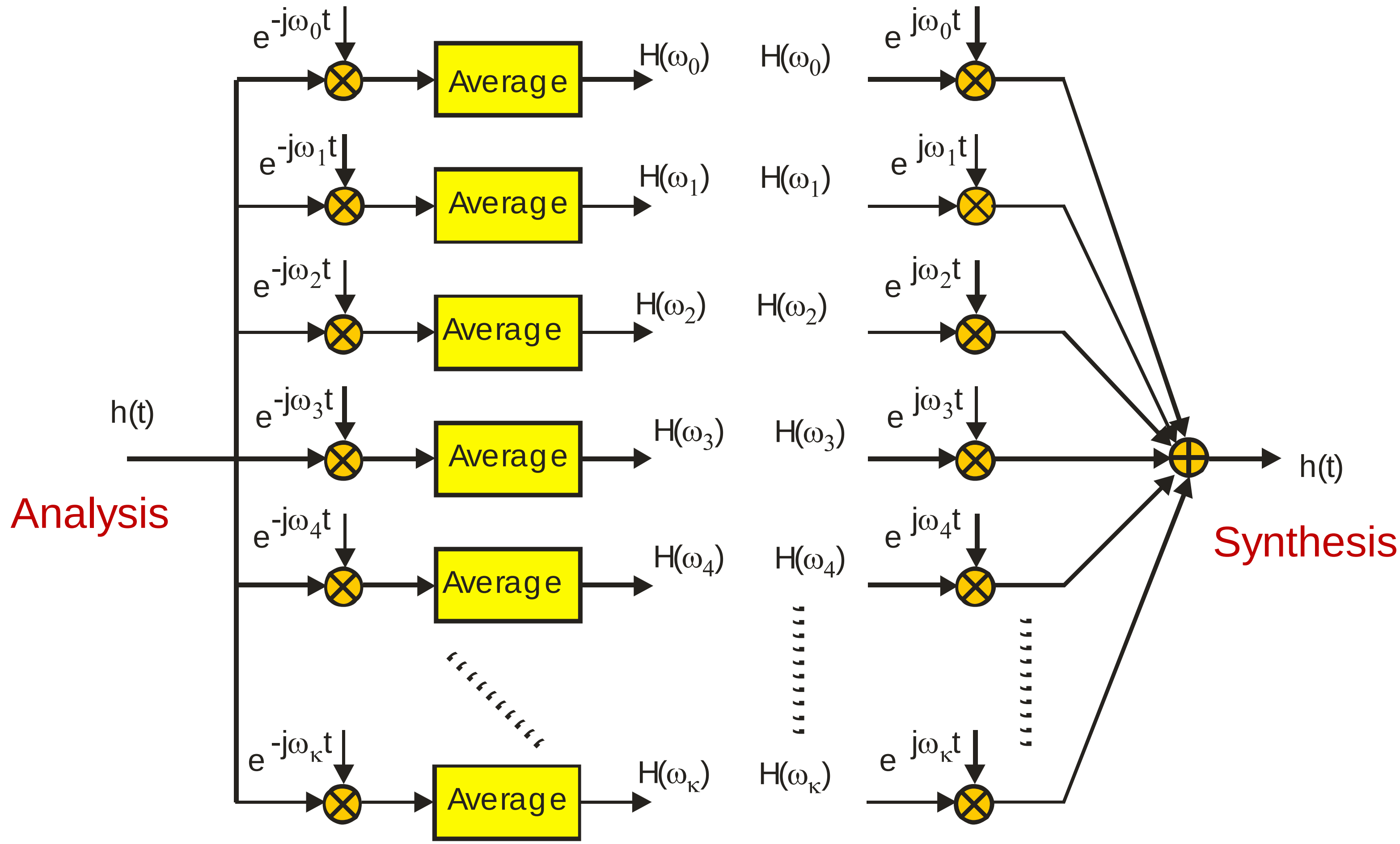
CONSTELLATION POINTS DISTRIBUTED OVER FREQUENCY INDEX



SEQUENTIAL SPECTRA SHOWING CONSTELLATION POINTS DISTRIBUTED OVER TIME-FREQUENCY INDICES

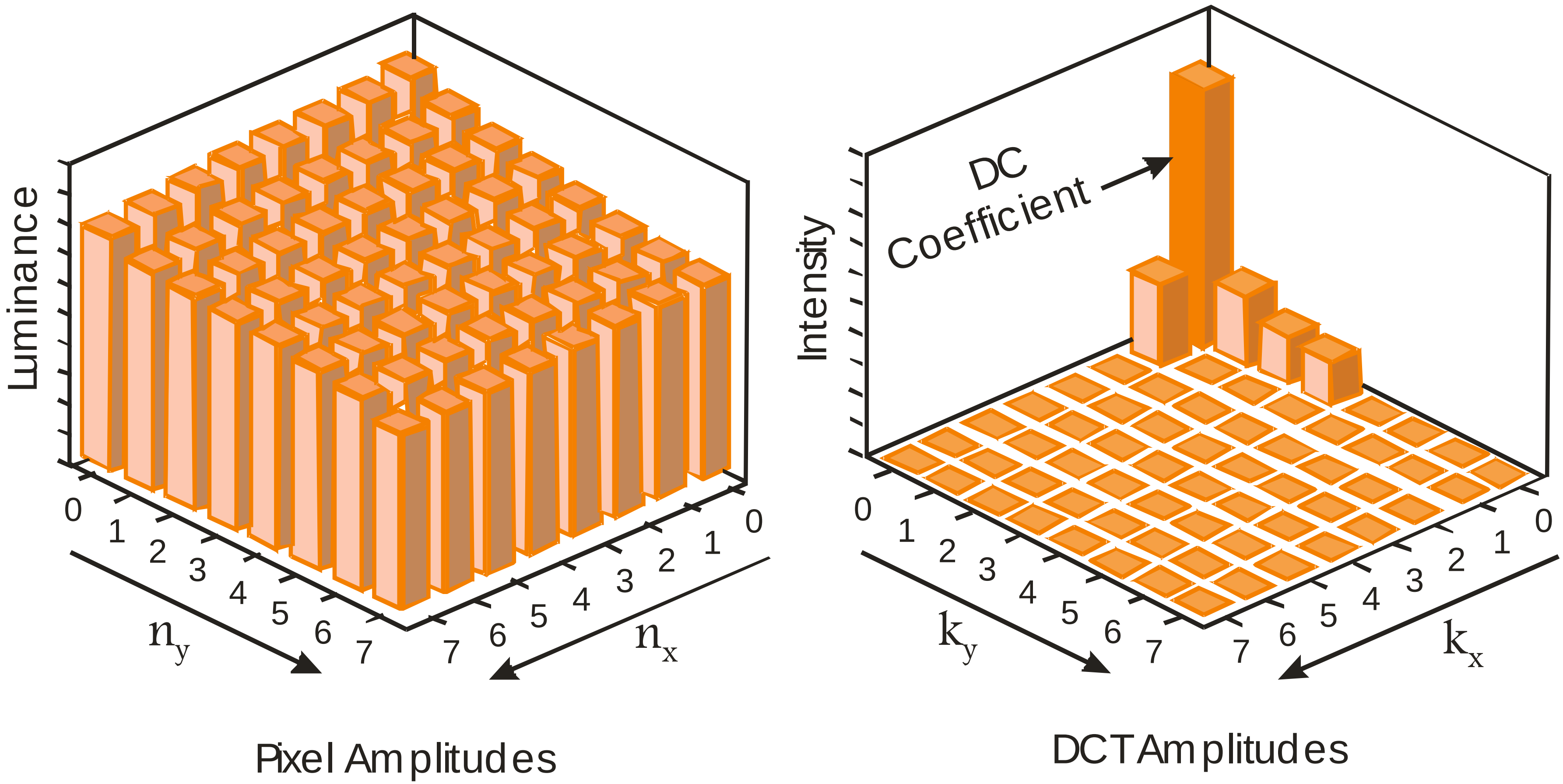


CASCADE FORWARD AND INVERSE FOURIER TRANSFORMS

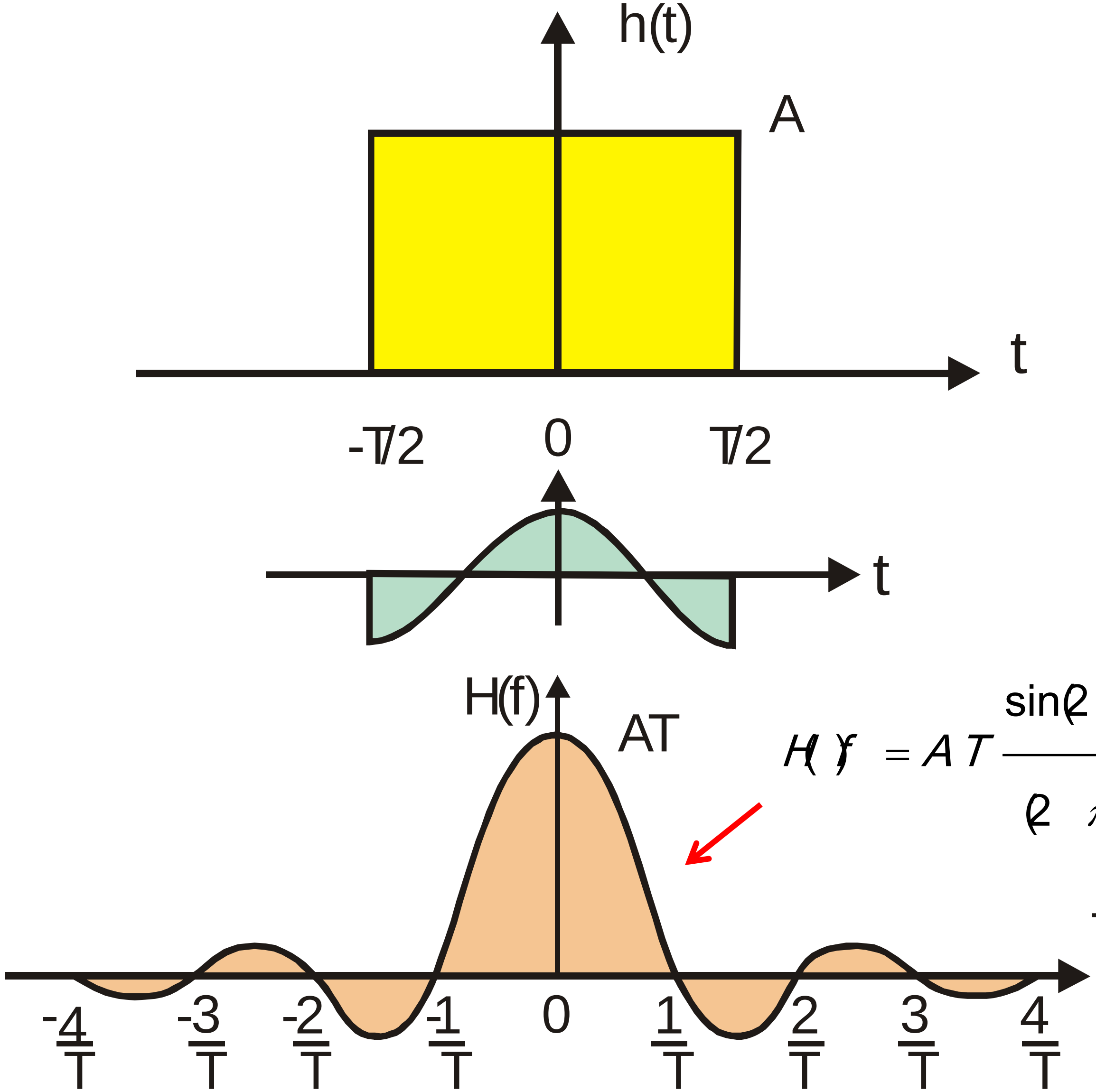


MPEG, MP3

MPEG DCT OF 8X8 PIXEL BLOCK

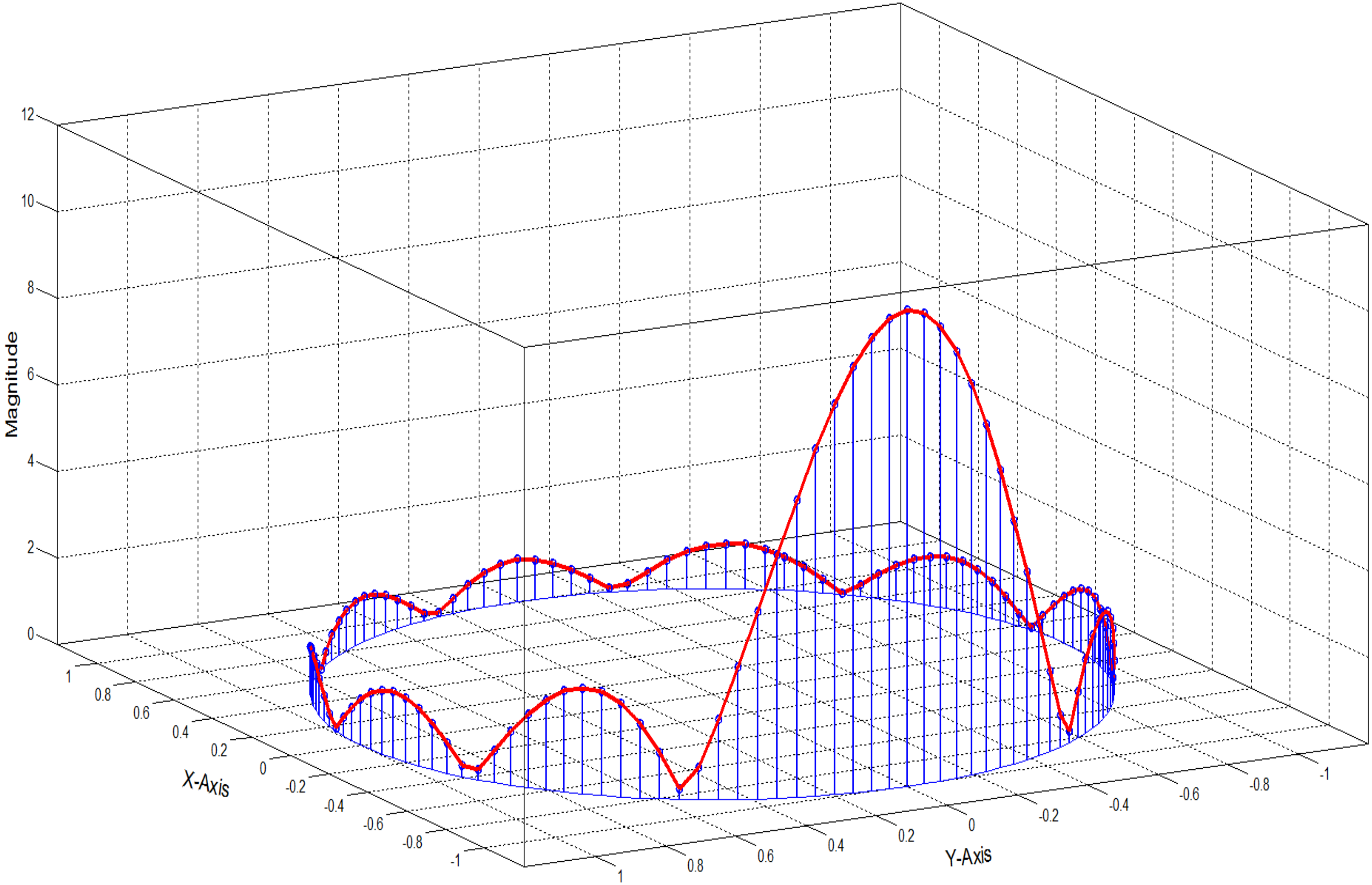


FOURIER TRANSFORM: RECTANGLE PULSE

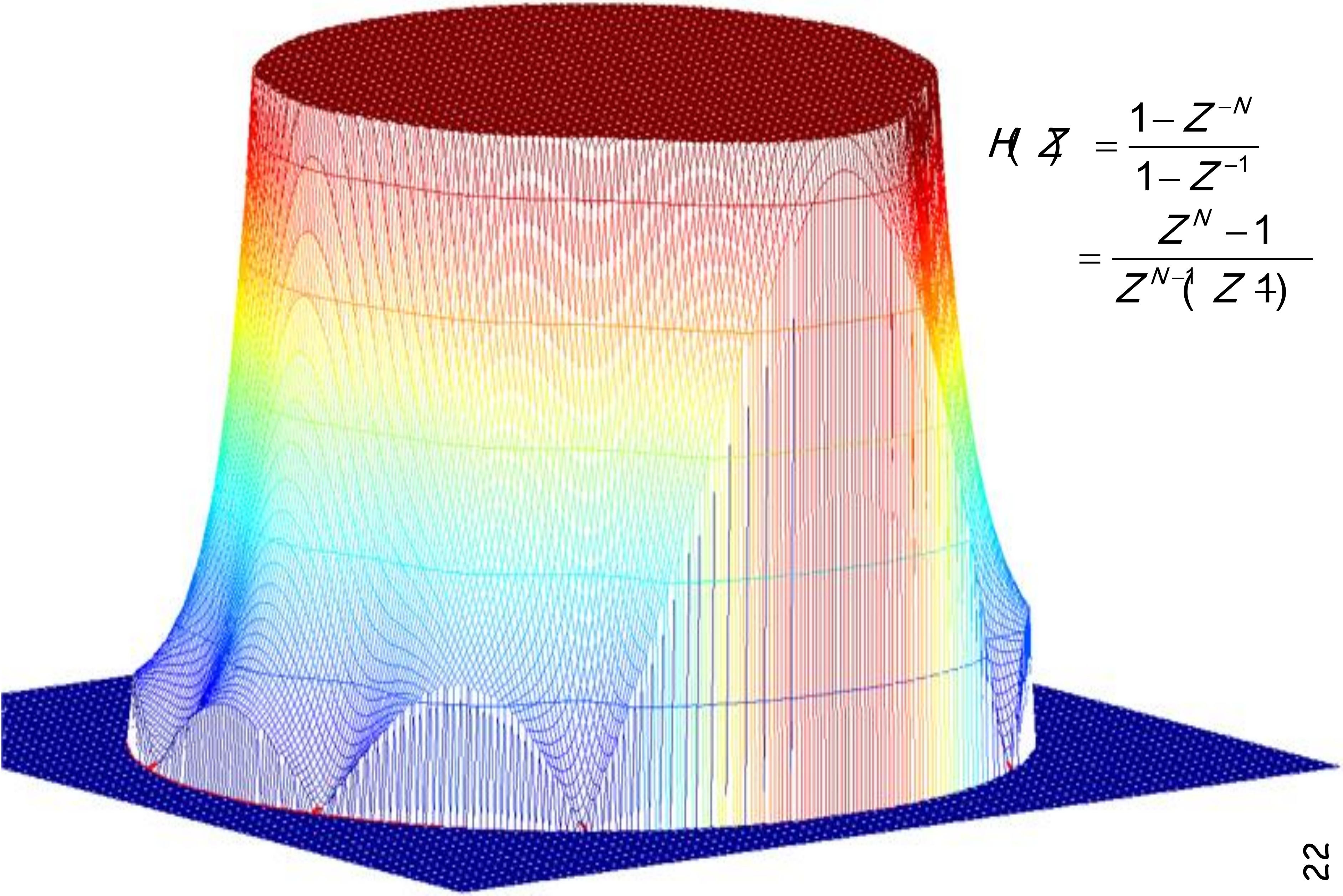


$$H(f) = AT\,\frac{\sin\left(2\pi f \frac{T}{2}\right)}{\left(2\pi f \frac{T}{2}\right)}$$

SIN(X)/X ON UNIT CIRCLE



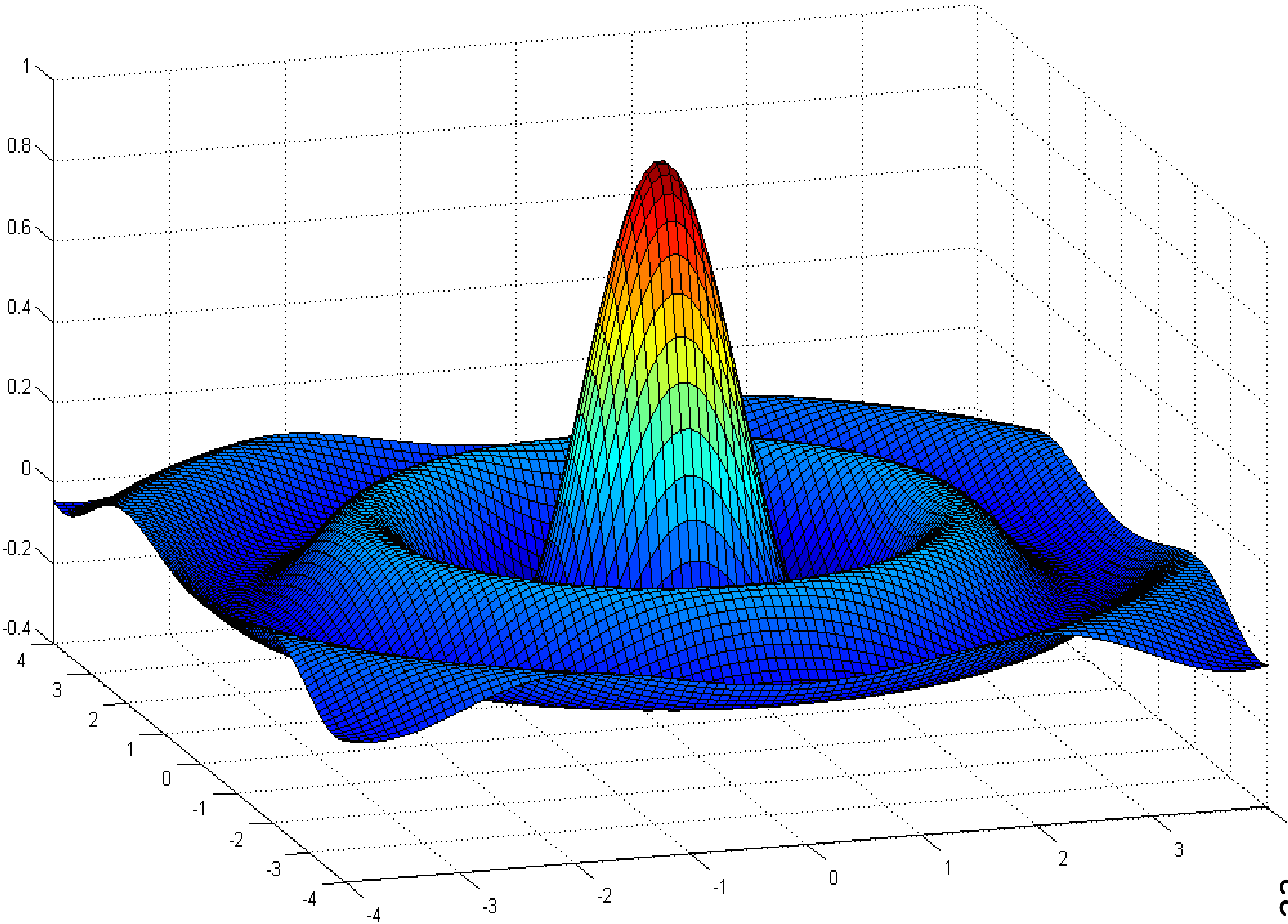
Sin(x)/x on Z-Plane



$$H(Z) = \frac{1 - Z^{-N}}{1 - Z^{-1}}$$

$$= \frac{Z^{N} - 1}{Z^{N-1}(Z - 1)}$$

2-DIMENSION SIN(X)/X

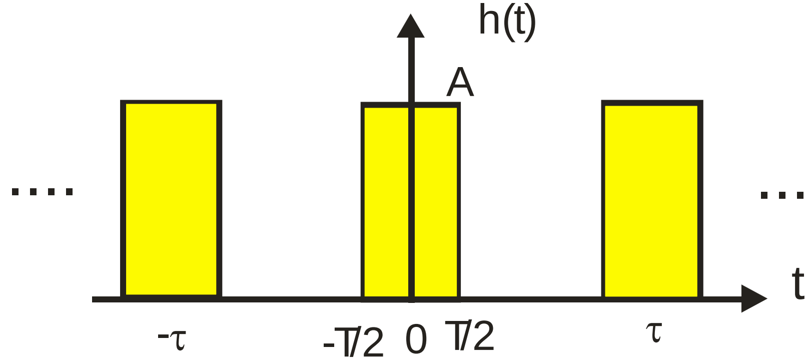


YOU'VE HEARD OF SIN(X)/X

HOW ABOUT SIN(X) OVER FRED?

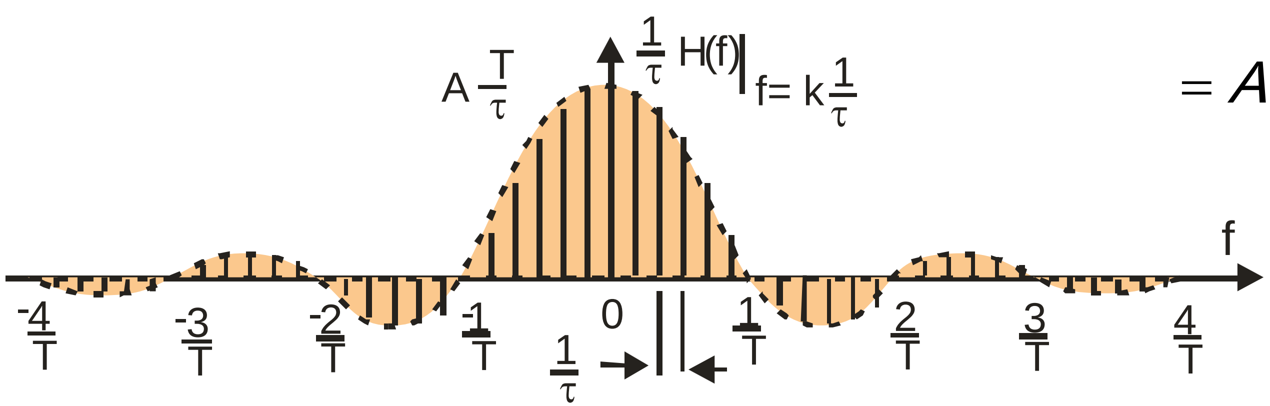
Street outside Asilomar Conference Grounds (Monterey, CA)

FOURIER SERIES OF PERIODIC RECTANGLE PULSES



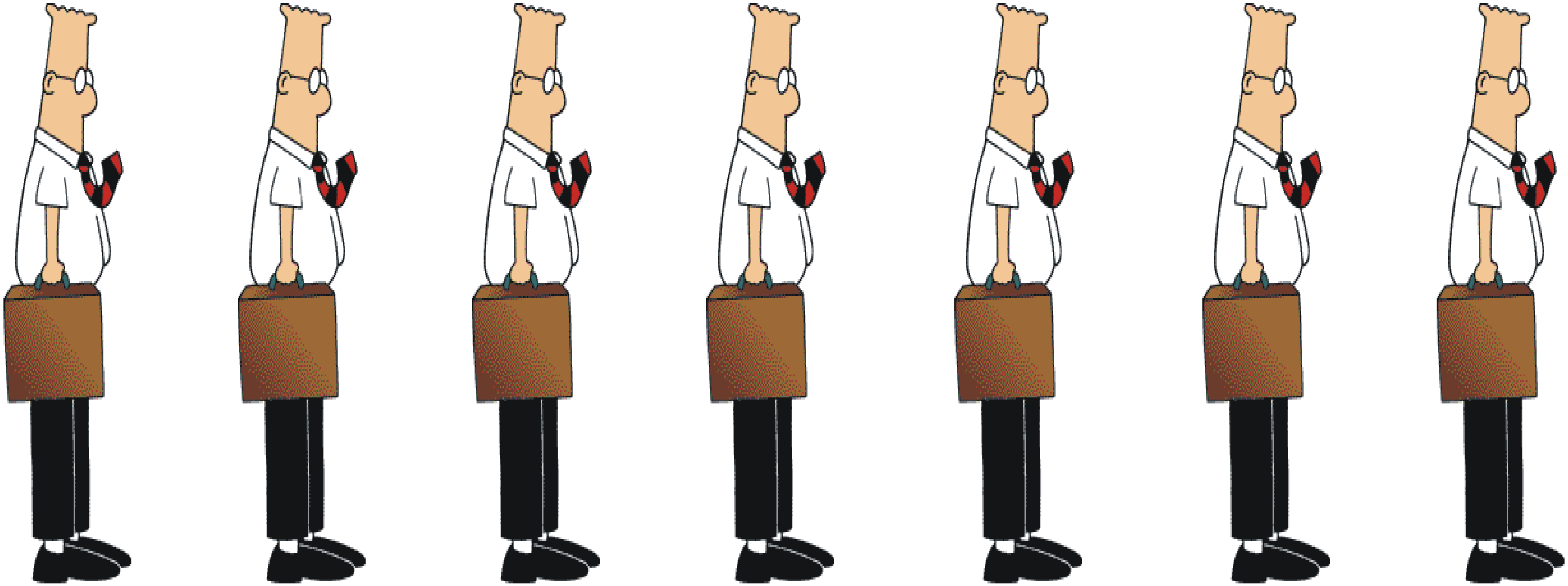
$$H(k) = \left(\frac{1}{\tau}\right) H(f)\Big|_{f=k\frac{1}{\tau}}$$

$$= A\frac{T}{\tau}\,\frac{\sin\left(\pi\frac{T}{2\tau}k\right)}{\left(\pi\frac{T}{2\tau}k\right)}$$



Uniform Samples of Spectrum
Describe Periodic Extension of Signal

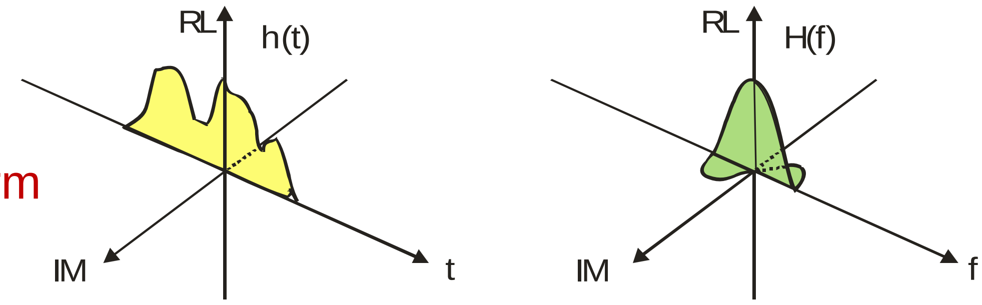
YOU'RE TELLING ME I'VE BECOME PERIODIC BECAUSE MY FOURIER TRANSFORM HAS BEEN SAMPLED?



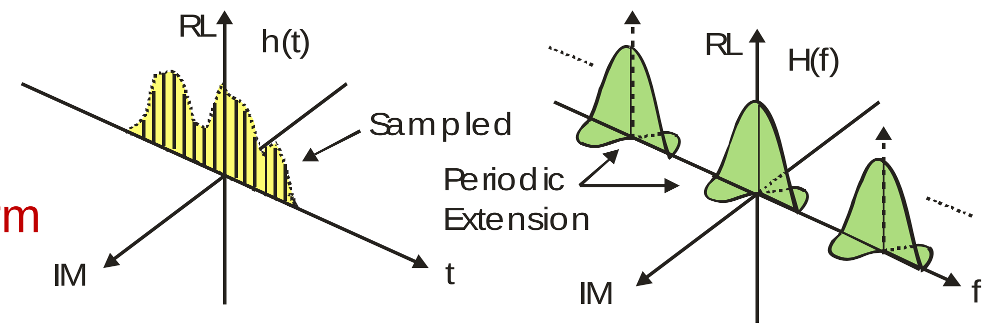
Yes indeed!

SAMPLING IN TIME AND FREQUENCY DOMAINS

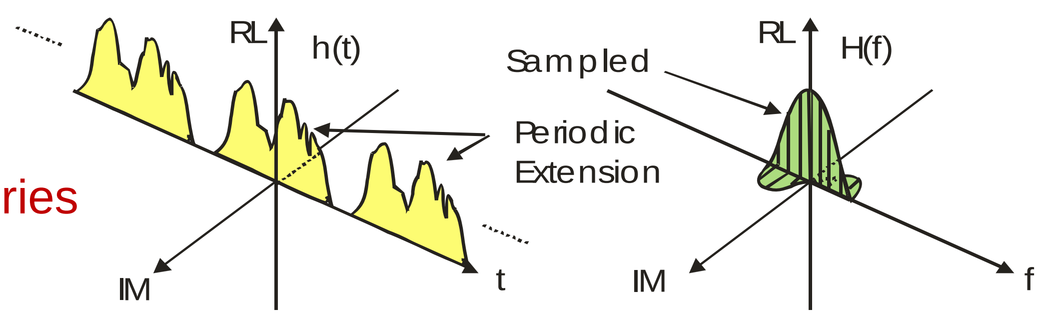
Fourier Transform



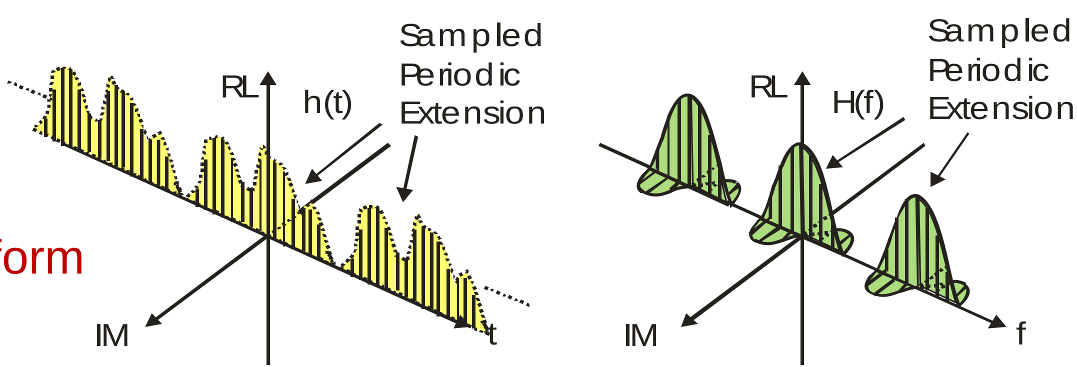
Sampled Data Fourier Transform



Fourier Series



Discrete Fourier Transform



SOME
FOURIER TRANSFORM
APPLICATIONS

DFT IS A COLLECTION OF MATCHED FILTERS FOR SINUSOIDS ALIGNED WITH THE FFT BASIS SET; (SINUSOIDS WITH INTEGER NUMBER OF CYCLES PER INTERVAL) IN ADDITIVE WHITE GAUSSIAN NOISE.

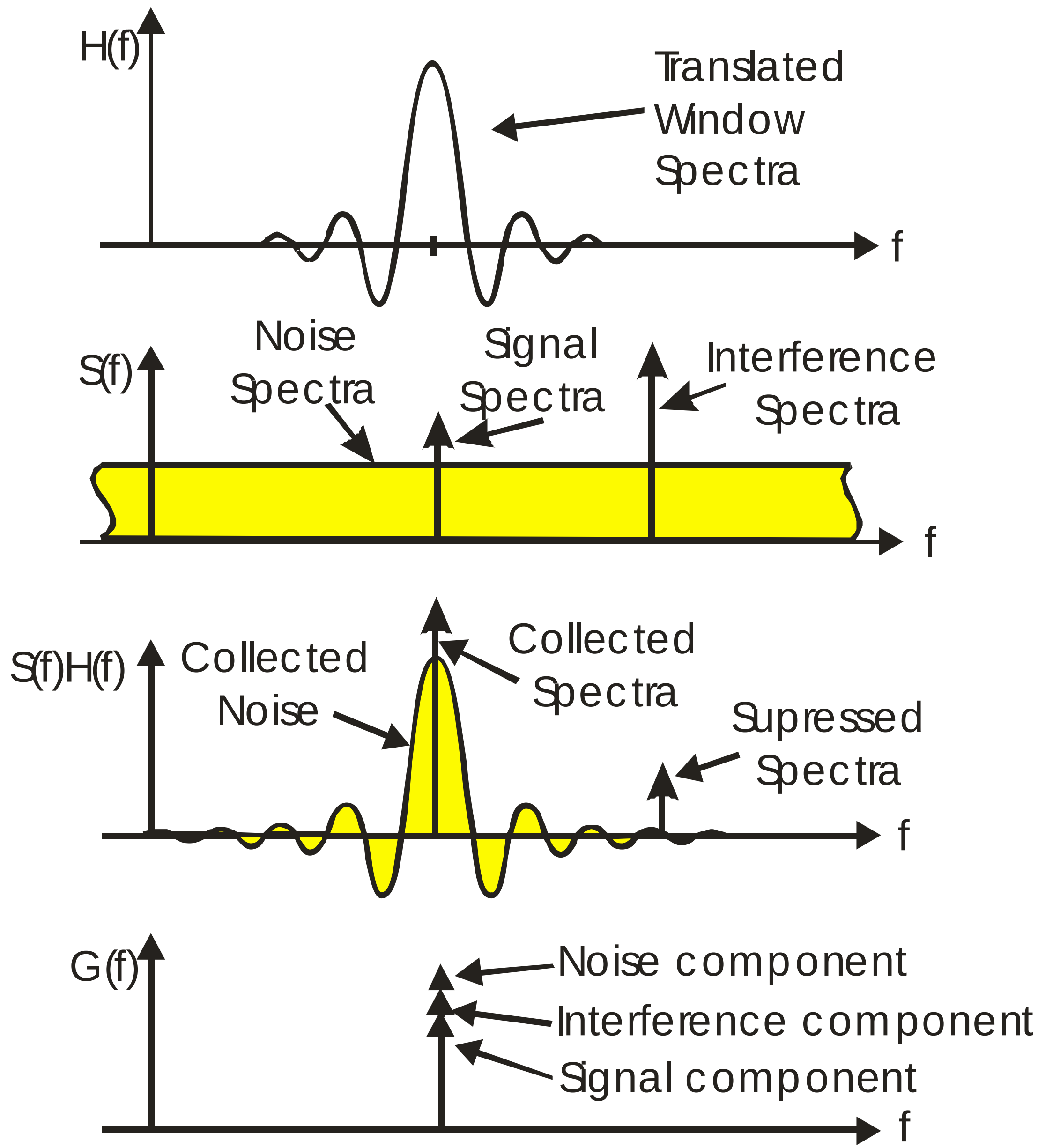
WE TEST AND TEACH HOW THE FFT PERFORMS WITH BASIS SET SINUSOIDS AND WITH OTHER HARMONICALLY RELATED PERIODIC SIGNALS (USUALLY WITHOUT ADDITIVE WHITE NOISE).

WE THEN USE THE FFT TO ANALYZE PERIODIC SIGNALS NOT ALIGNED WITH THE FFT BASIS SET OR NON PERIODIC RANDOM SIGNALS IN NON WHITE (COLORED) RANDOM NOISE.

IT SHOULD COME AS NO SURPRISE THAT WE HAVE LIMITED UNDERSTANDING OF THE FFT'S PERFORMANCE WHEN PROCESSING THIS WIDER RANGE OF INPUT SIGNALS.

PROPERLY DESIGNED COMMUNICATION SIGNALS HAVE THE SAME STATISTICS AS BAND LIMITED WHITE NOISE (MAY HAVE OTHER PROPERTIES TOO!)

SPECTRAL CONTENT OF SIGNAL



DFT OF SIGNALS, INTERFERENCE AND NOISE

$$d(n) = A_{Sig}\, e^{j\theta_{Sig}}\, e^{j\frac{2\pi}{N} n k_{Sig}} + A_{Intf}\, e^{j\theta_{Intf}}\, e^{j\frac{2\pi}{N} n k_{Intf}} + N(n)$$

$$H(k) = \sum_{n=0}^{N-1} d(n) w(n) e^{-j\frac{2\pi}{N} n k}$$

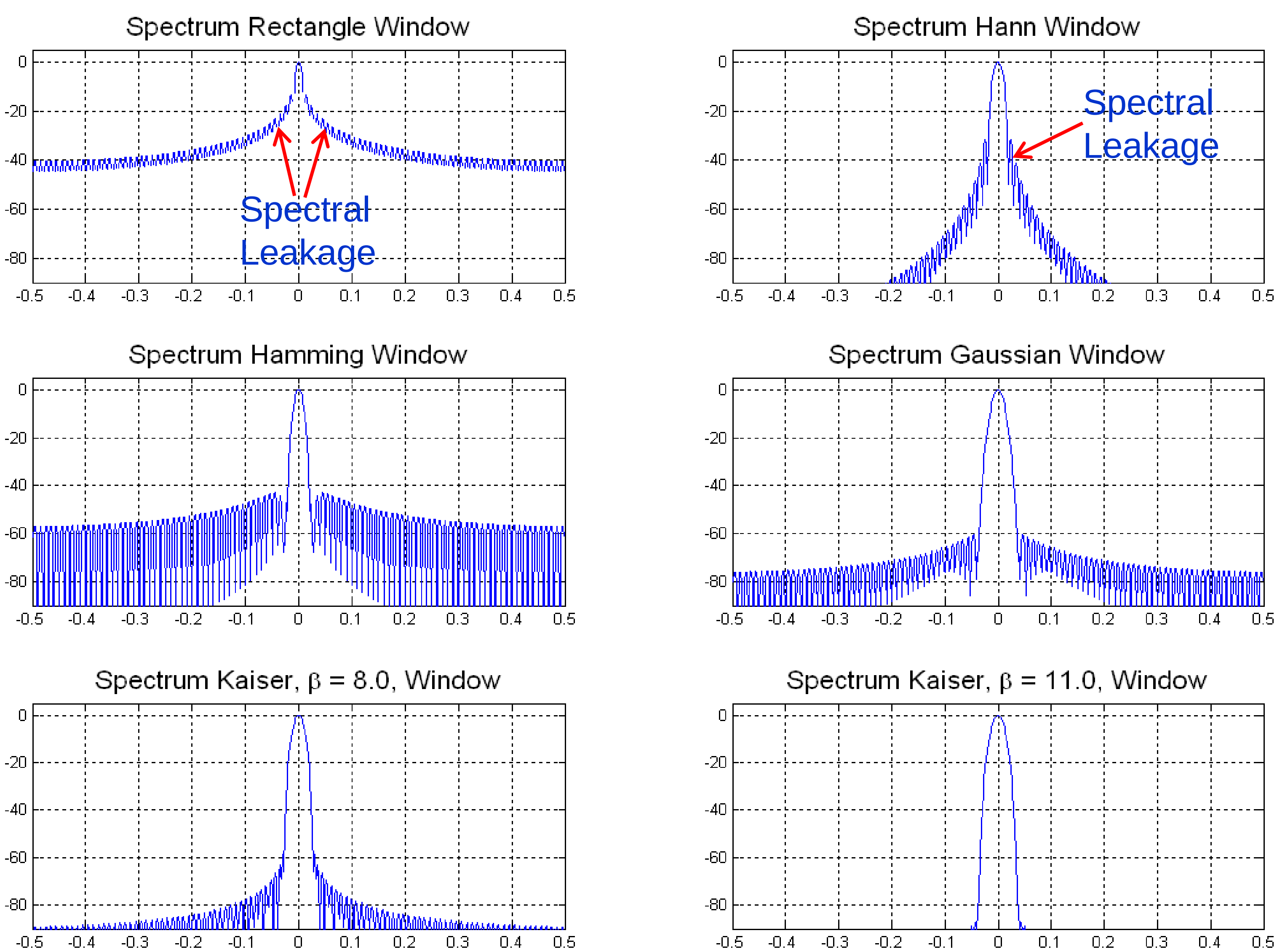
$$= H_{Sig}(k) + H_{Intf}(k) + H_{Noise}(k)$$

$$H_{Sig}(k) = A_{Sig}\, e^{j\theta_{Sig}} \sum_{n=0}^{N-1} w(n) e^{-j\frac{2\pi}{N} n (k-k_{sig})}$$

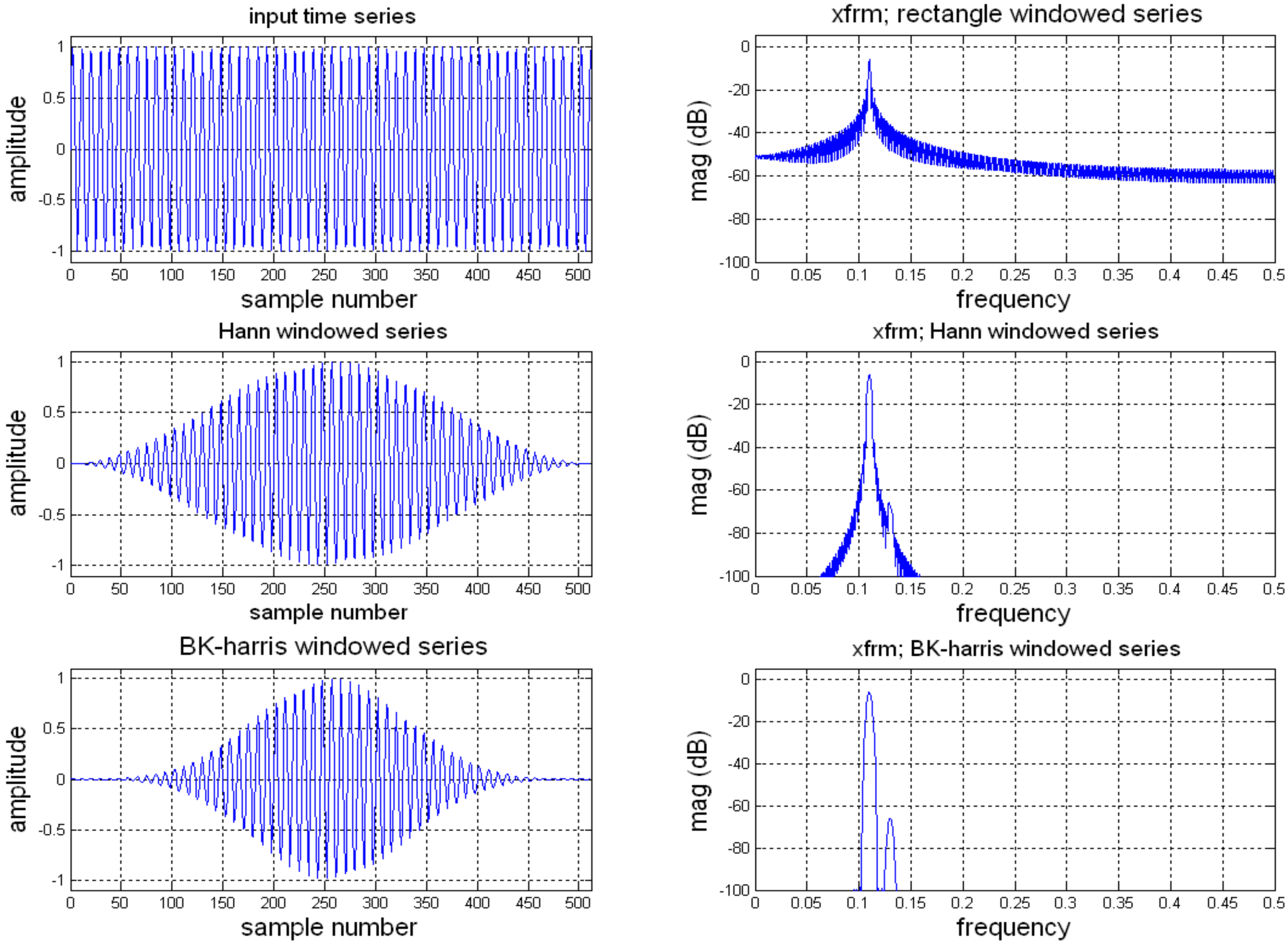
$$H_{Intf}(k) = A_{Intf}\, e^{j\theta_{Intf}} \sum_{n=0}^{N-1} w(n) e^{-j\frac{2\pi}{N} n (k-k_{Int})}$$

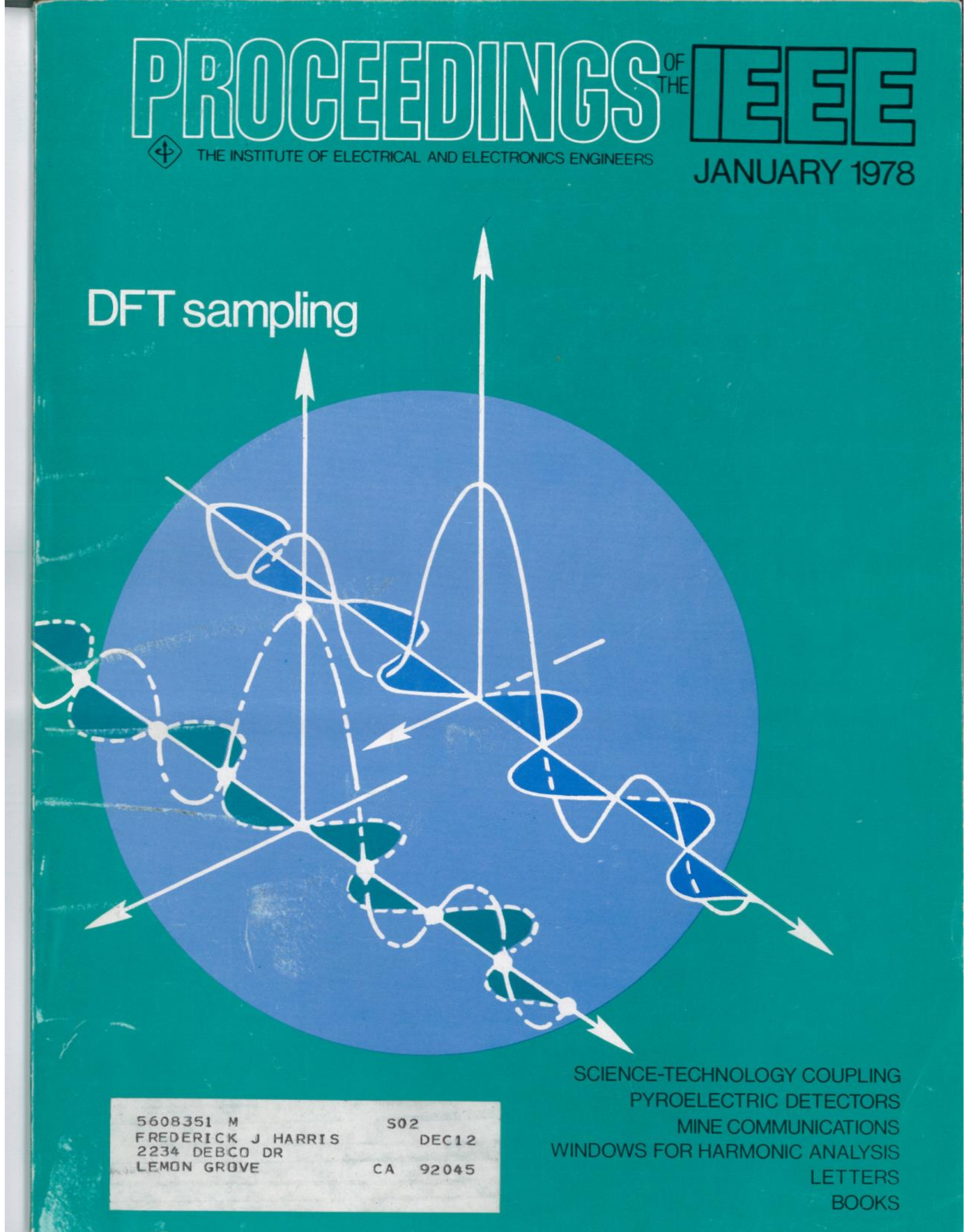
$$H_{Noise}(k) = \sum_{n=0}^{N-1} N(n)\, w(n) e^{-j\frac{2\pi}{N} n (k-k_{Int})}$$

SPECTRA OF CLASSIC WINDOWS



SPECTRUM ANALYSIS
NEED WINDOWS TO SUPPRESS PROCESSING ARTIFACTS
(PROCESS SINUSOIDS WITH NON-INTEGER NUMBER OF CYCLES IN PROCESSING INTERVAL)




PROCEEDINGS OF THE IEEE
THE INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS
JANUARY 1978
DFT sampling
5608351 M S02
FREDERICK J HARRIS DEC12
2234 DEBCO DR
LEMON GROVE CA 92045
SCIENCE-TECHNOLOGY COUPLING
PYROELECTRIC DETECTORS
MINE COMMUNICATIONS
WINDOWS FOR HARMONIC ANALYSIS
LETTERS
BOOKS

On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform

FREDRIC J. HARRIS, MEMBER, IEEE

Abstract—This paper makes available a concise review of data windows and their affect on the detection of harmonic signals in the presence of broad-band noise, and in the presence of nearby strong harmonic interference. We also call attention to a number of common errors in the application of windows when used with the fast Fourier transform. This paper includes a comprehensive catalog of data windows along with their significant performance parameters from which the different windows can be compared. Finally, an example demonstrates the use and value of windows to resolve closely spaced harmonic signals characterized by large differences in amplitude.

I. Introduction

THERE IS MUCH signal processing devoted to detection and estimation. Detection is the task of determining if a specific signal set is present in an observation, while estimation is the task of obtaining the values of the parameters describing the signal. Often the signal is complicated or is corrupted by interfering signals or noise. To facilitate the detection and estimation of signal sets, the observation is decomposed by a basis set which spans the signal space [1]. For many problems of engineering interest, the class of signals being sought are periodic which leads quite naturally to a decomposition by a basis consisting of simple periodic functions, the sines and cosines. The classic Fourier transform is the mechanism by which we are able to perform this decomposition.

By necessity, every observed signal we process must be of finite extent. The extent may be adjustable and selectable, but it must be finite. Processing a finite-duration observation imposes interesting and interacting considerations on the harmonic analysis. These considerations include detectability of tones in the presence of nearby strong tones, resolvability of similar-strength nearby tones, resolvability of shifting tones, and biases in estimating the parameters of any of the aforementioned signals.

For practicality, the data we process are N uniformly spaced samples of the observed signal. For convenience, N is highly composite, and we will assume N is even. The harmonic estimates we obtain through the discrete Fourier transform (DFT) are N uniformly spaced samples of the associated periodic spectra. This approach is elegant and attractive when the processing scheme is cast as a spectral decomposition in an N-dimensional orthogonal vector space [2]. Unfortunately, in many practical situations, to obtain meaningful results this elegance must be compromised. One such compromise consists of applying windows to the sampled data set, or equivalently, smoothing the spectral samples.

The two operations to which we subject the data are sampling and windowing. These operations can be performed in either order. Sampling is well understood, windowing is less so, and sampled windows for DFT's significantly less so! We will address the interacting considerations of window selection in harmonic analysis and examine the special considerations related to sampled windows for DFT's.

II. Harmonic Analysis of Finite-Extent Data and the DFT

Harmonic analysis of finite-extent data entails the projection of the observed signal on a basis set spanning the observation interval [1], [3]. Anticipating the next paragraph, we define T seconds as a convenient time interval and NT seconds as the observation interval. The sines and cosines with periods equal to an integer submultiple of NT seconds form an orthogonal basis set for continuous signals extending over NT seconds. These are defined as

$$\left.\begin{matrix}\cos\left[\dfrac{2\pi}{NT}kt\right]\\[2ex] \sin\left[\dfrac{2\pi}{NT}kt\right]\end{matrix}\right\}\quad \begin{matrix}k=0,1,\cdots,N-1,N,N+1,\cdots\\[2ex] 0\leqslant t<NT.\end{matrix}\tag{1}$$

We observe that by defining a basis set over an ordered index k, we are defining the spectrum over a line (called the frequency axis) from which we draw the concepts of bandwidth and of frequencies close to and far from a given frequency (which is related to resolution).

For sampled signals, the basis set spanning the interval of NT seconds is identical with the sequences obtained by uniform samples of the corresponding continuous spanning set up to the index $N/2$,

$$\left.\begin{matrix}\cos\left[\dfrac{2\pi}{NT}knT\right]=\cos\left[\dfrac{2\pi}{N}kn\right]\\[2ex] \sin\left[\dfrac{2\pi}{NT}knT\right]=\sin\left[\dfrac{2\pi}{N}kn\right]\end{matrix}\right\}\quad \begin{matrix}k=0,1,\cdots,N/2\\[2ex] n=0,1,\cdots,N-1.\end{matrix}\tag{2}$$

We note here that the trigonometric functions are unique in

Manuscript received September 10, 1976; revised April 11, 1977 and September 1, 1977. This work was supported by Naval Undersea Center (now Naval Ocean Systems Center) Independent Exploratory Development Funds.

The author is with the Naval Ocean Systems Center, San Diego, CA, and the Department of Electrical Engineering, School of Engineering, San Diego State University, San Diego, CA 92182.

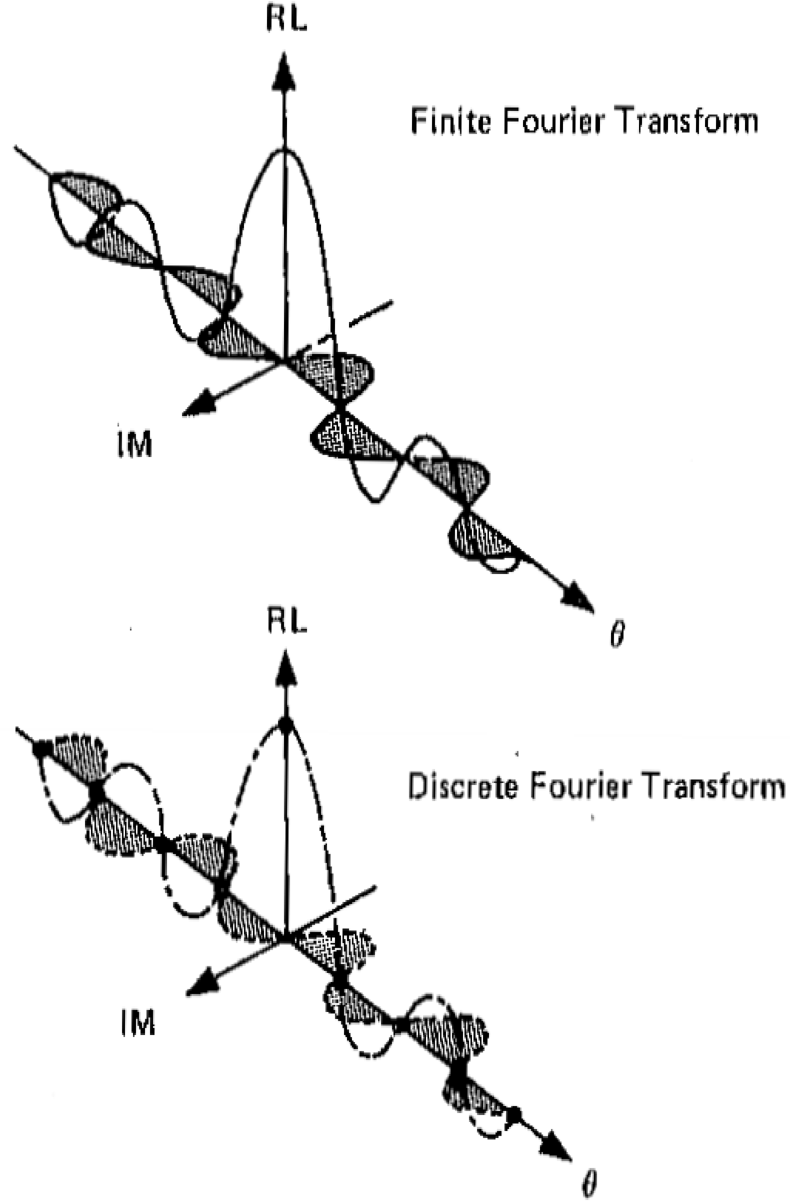
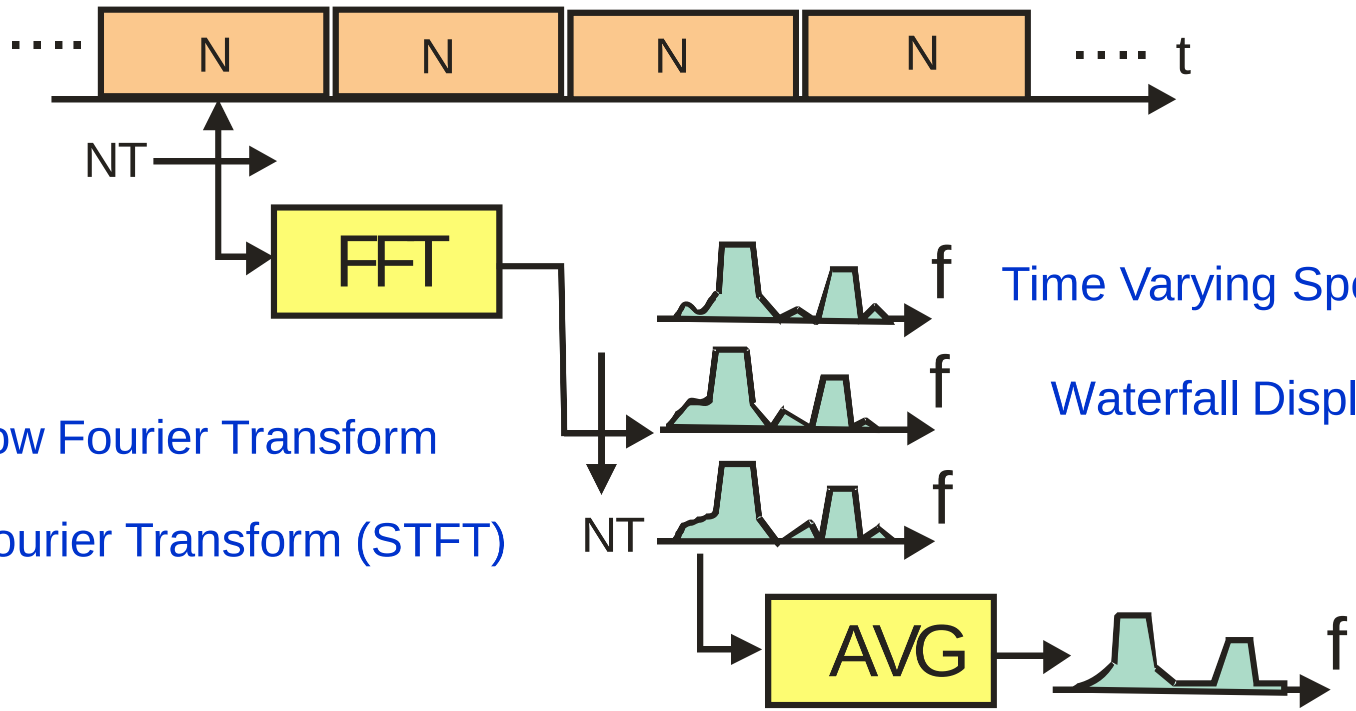


Fig. 3. DFT sampling of finite Fourier transform of a DFT even sequence.

Fredric J. Harris (M'61) was born in Brooklyn, NY, in 1940. He received the B.E.E. degree from the Polytechnic Institute of Brooklyn, Brooklyn, NY, in 1961, the M.S.E.E. degree from San Diego State University, San Diego, CA, in 1967, and is completing requirements for the Ph.D. degree at the University of California at San Diego, La Jolla.

He is currently an Associate Professor in the School of Engineering at San Diego State University, and has been a faculty member there since 1967. On leave of absence from the University, he currently holds a part time position with the Naval Ocean Systems Center in San Diego where he performs research on digital signal processing. He also consults for a number of companies in the San Diego area and offers seminars in signal processing and in the fast Fourier transform. His interests include digital signal processing, adaptive filtering, and communication theory.

ENSEMBLE AVERAGE TO IMPROVE STATISTICS OF SPECTRAL ESTIMATE



Time Varying Spectra

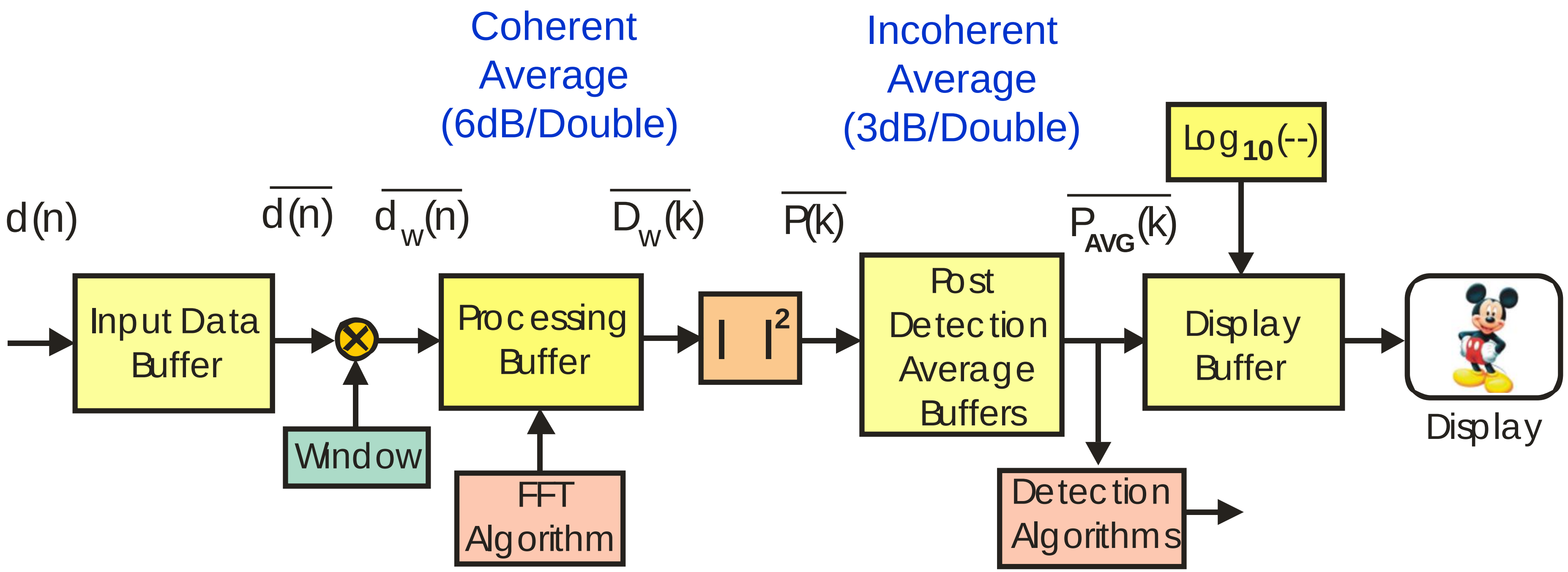
Waterfall Display

Sliding Window Fourier Transform

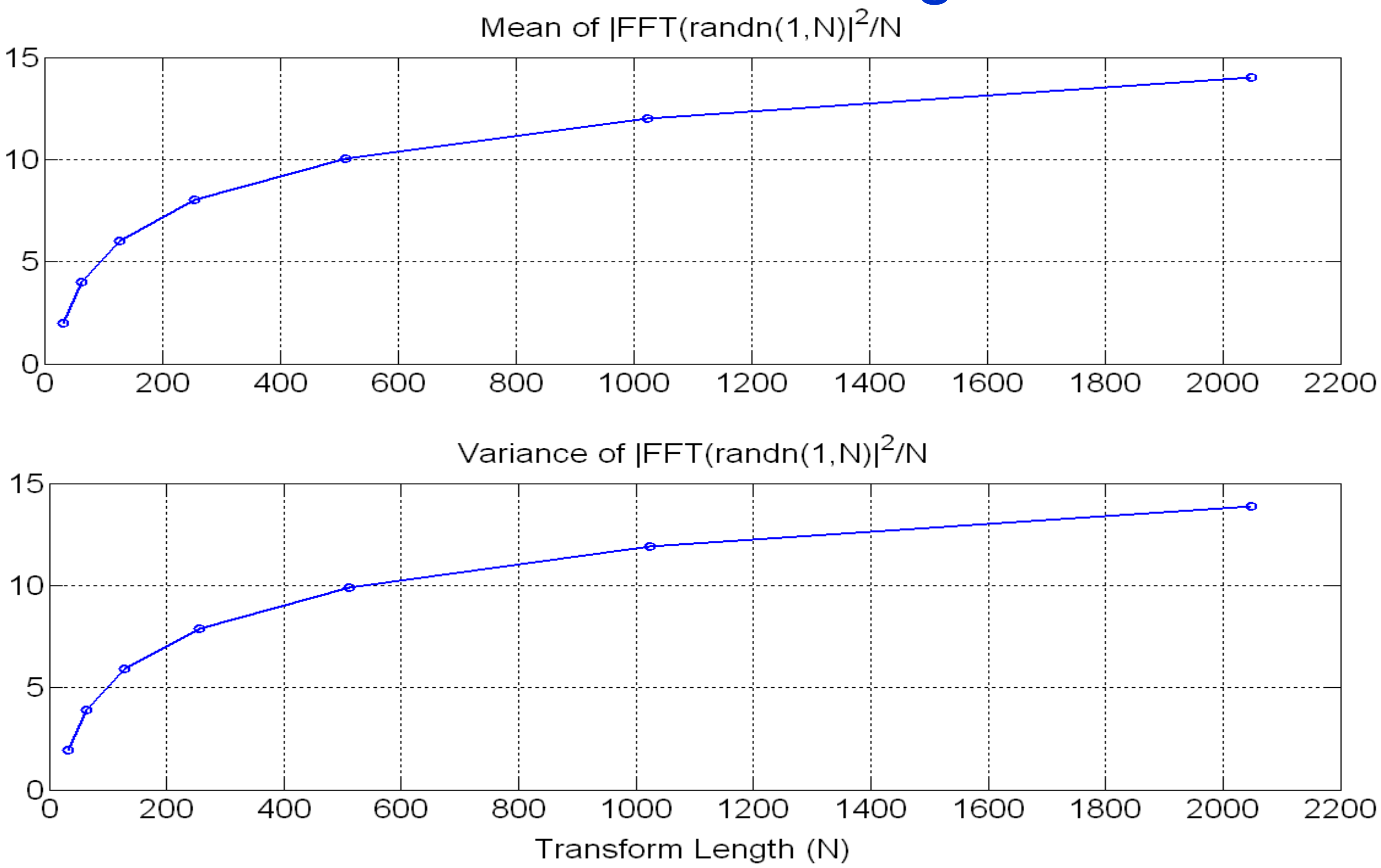
Short Time Fourier Transform (STFT)

Ensemble Average Spectrum

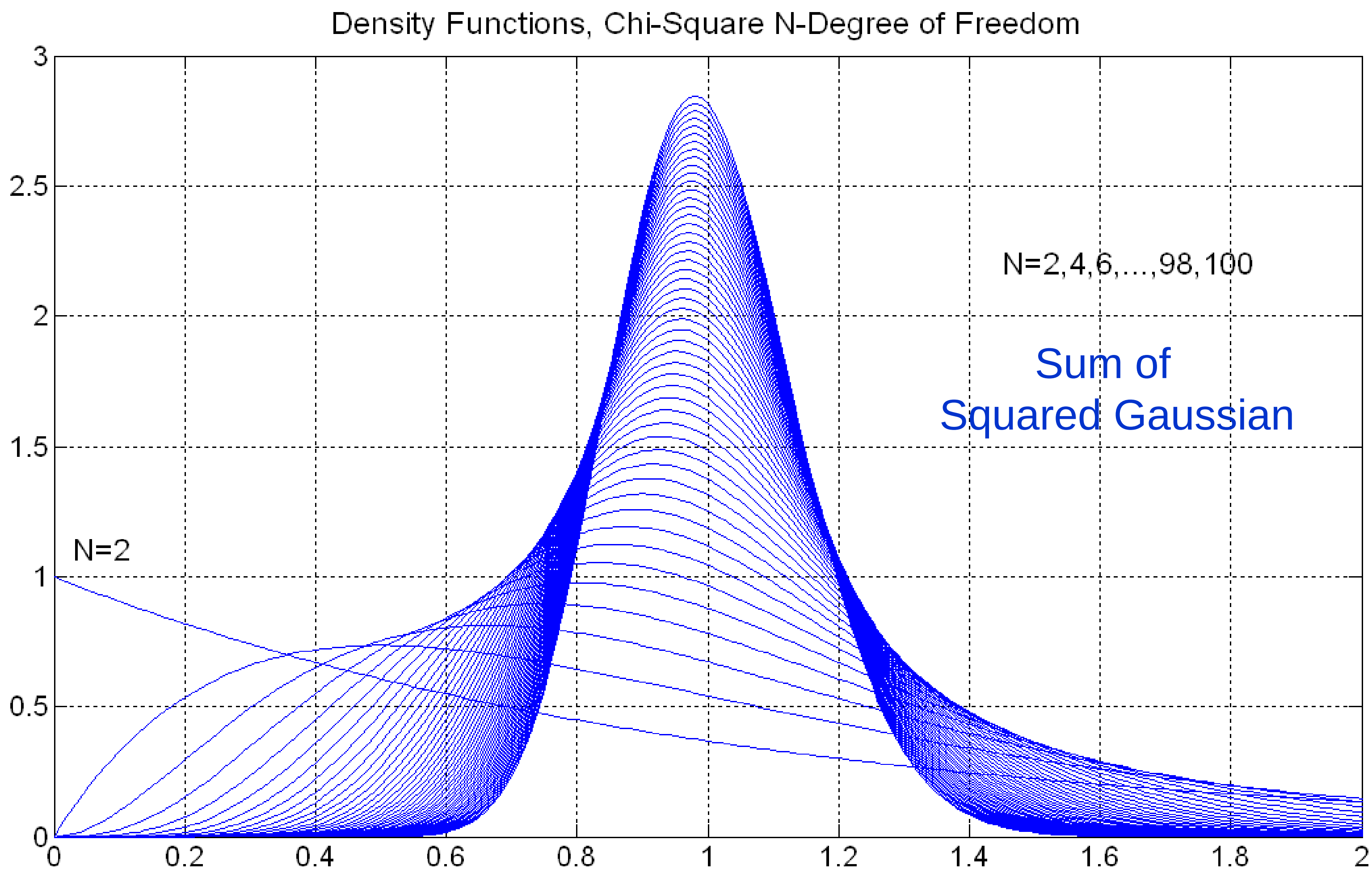
POWER SPECTRUM ESTIMATION OF RANDOM SIGNALS WITH FFT



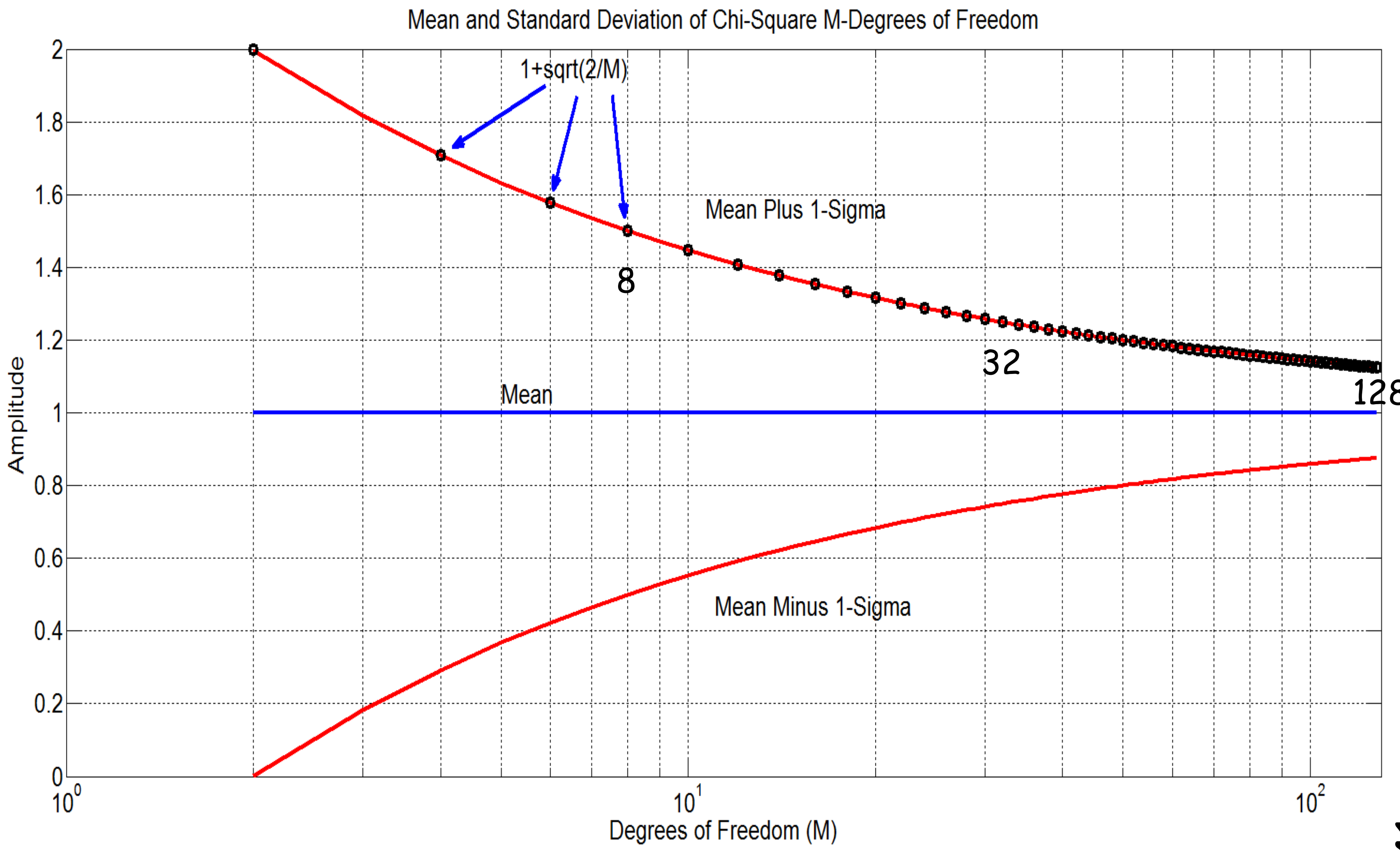
Random Signal Power Spectral Estimation Equal Standard Deviation and Mean Both Increase with FFT Length



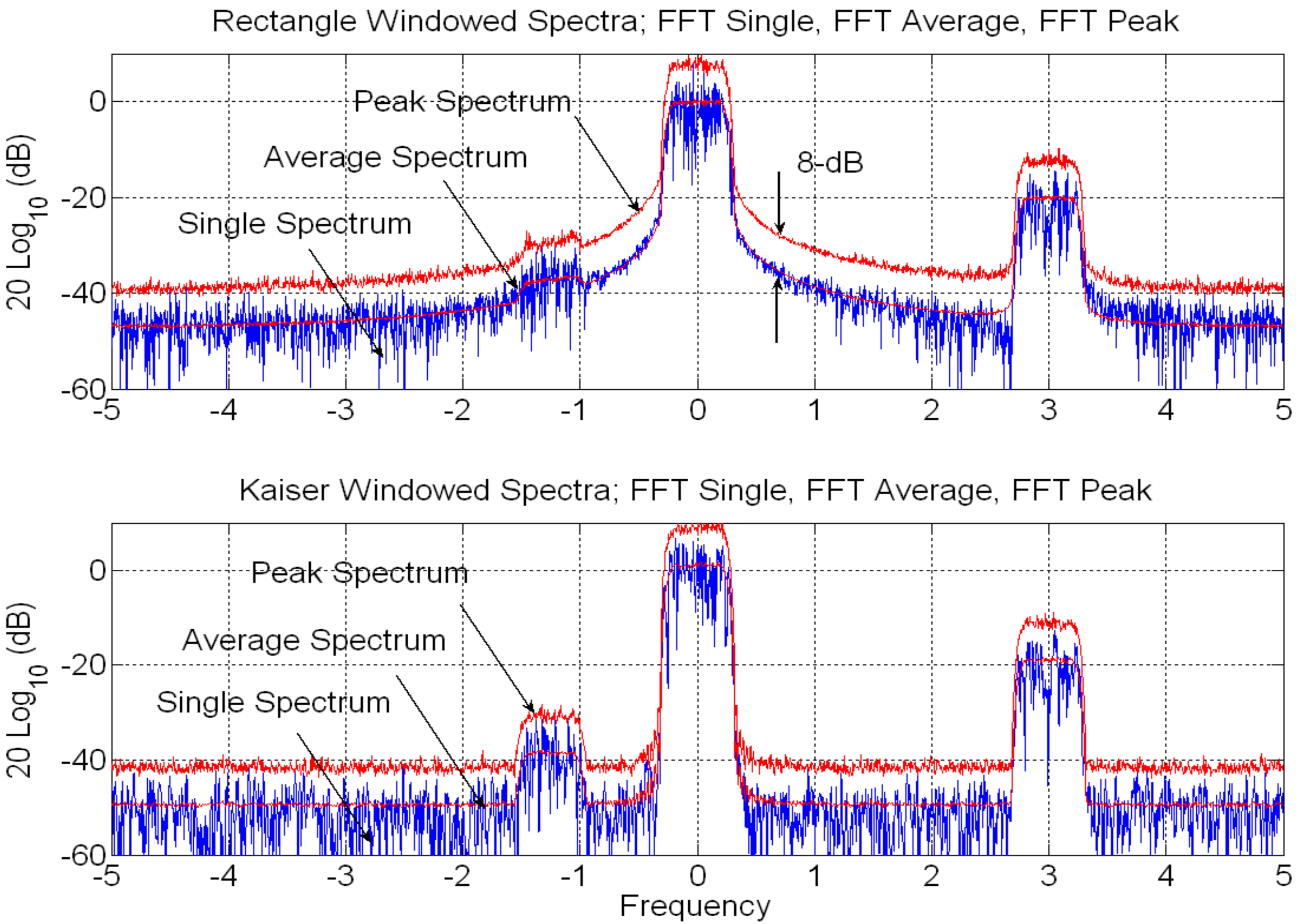
DENSITY FUNCTIONS,
CHI-SQUARE, N-DEGREES OF FREEDOM



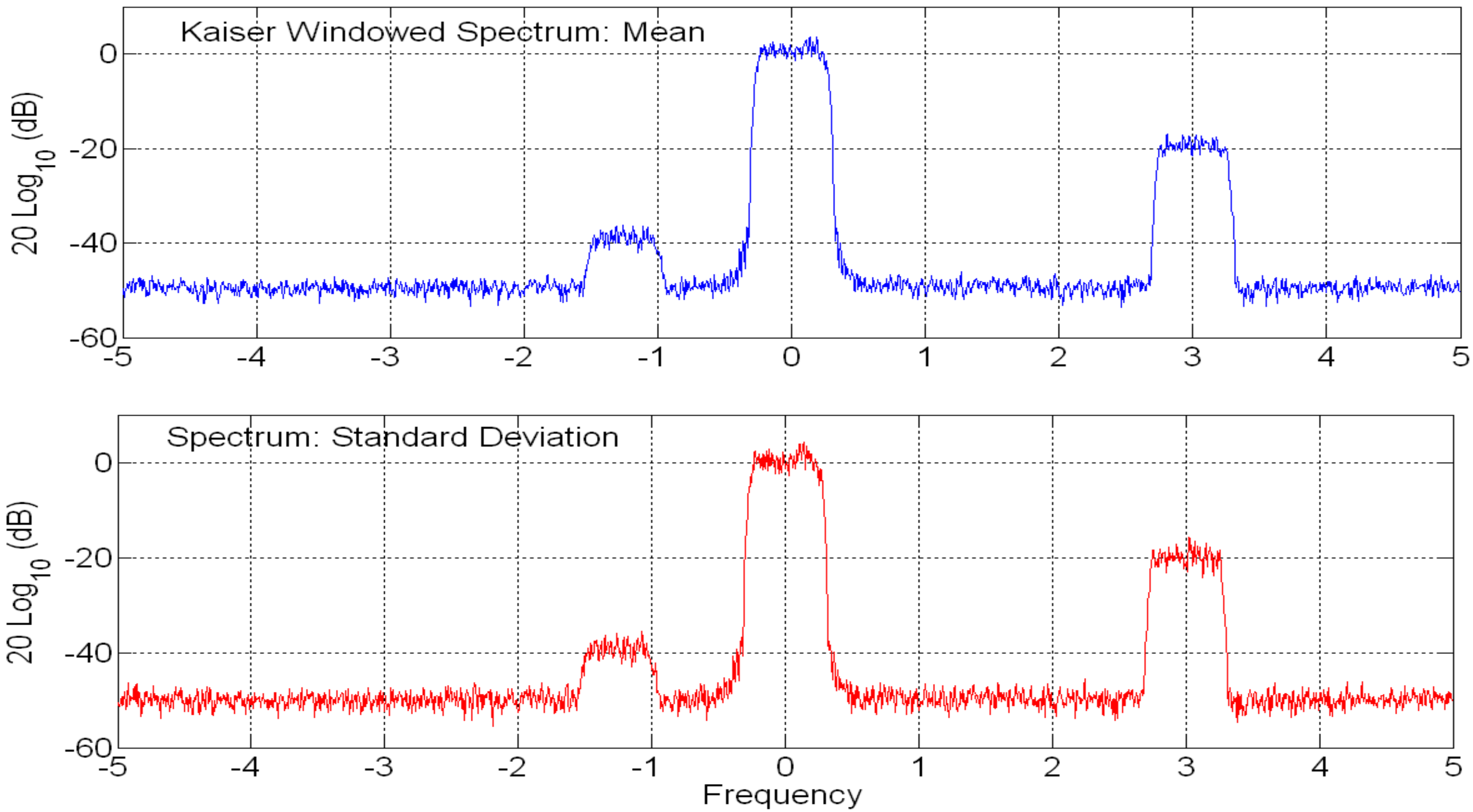
MEAN AND STANDARD DEVIATION: CHI-SQUARE, N-DEGREES OF FREEDOM



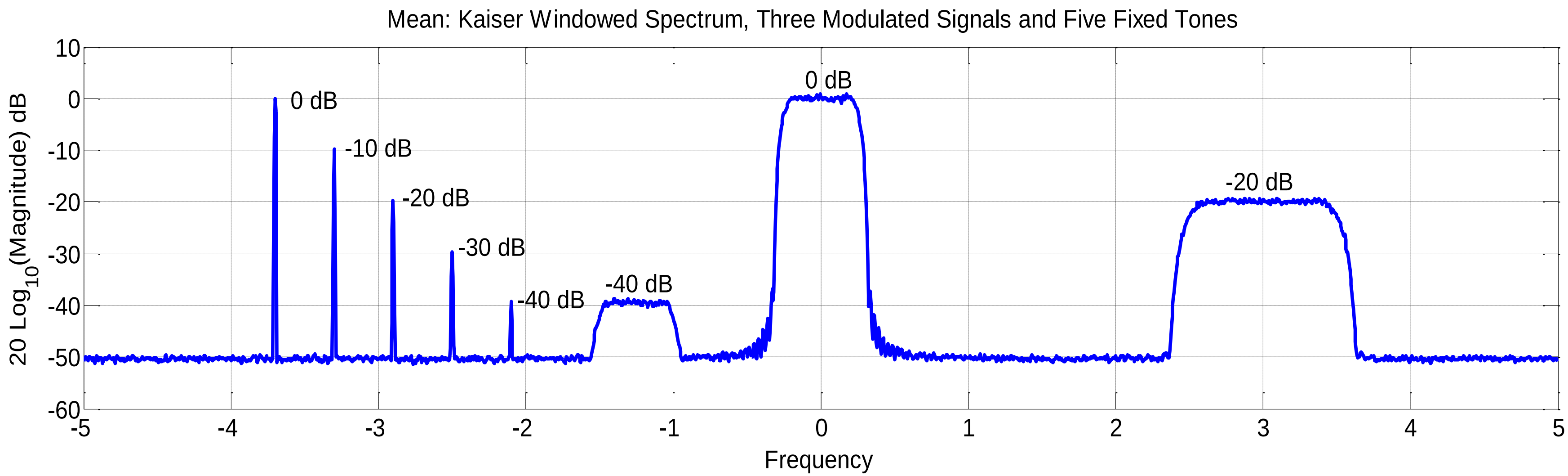
SPECTRA: SINGLE (RAW), AVERAGE, AND PEAK

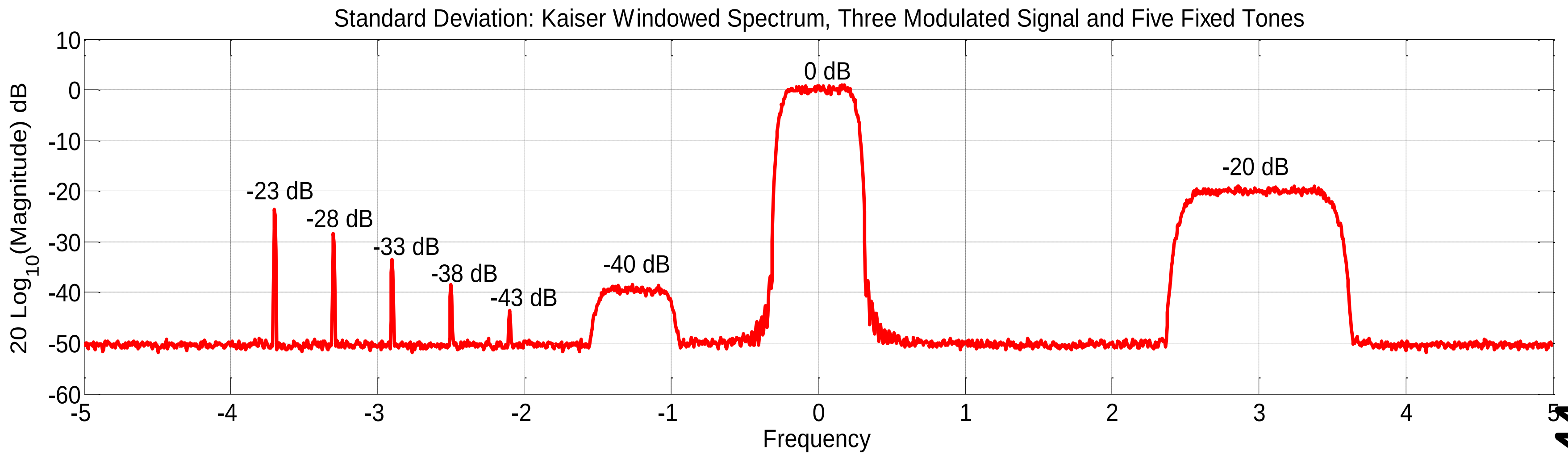


WEIRD PROPERTY OF TRANSFORM: (AN INCONSISTENT ESTIMATOR) SAMPLE MEAN AND STANDARD DEVIATION



Property of Transform: Tones and Noise Like Signals Have Different Sample Mean and Standard Deviation



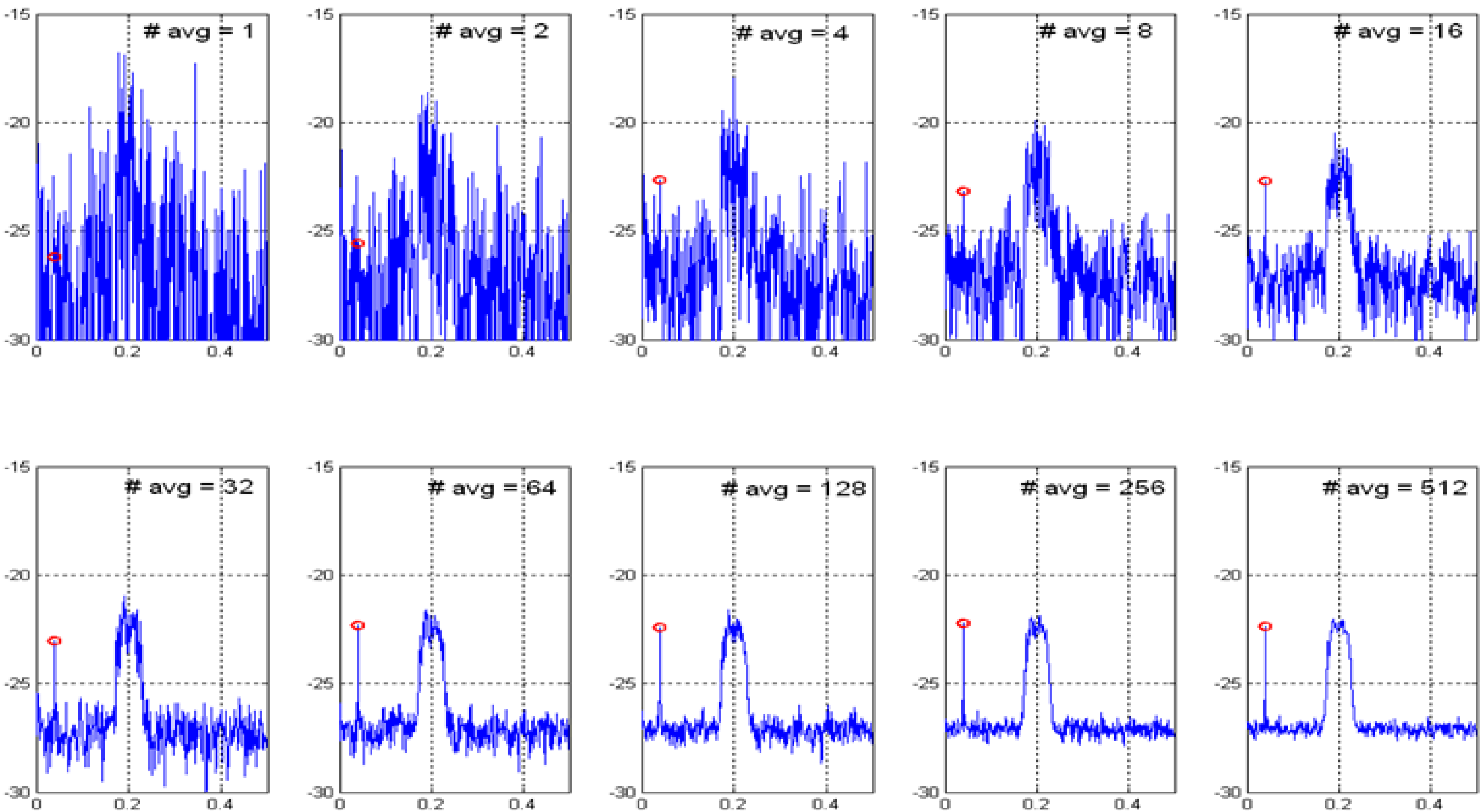


SPECTRAL ESTIMATION IN NOISE (NEED FOR ENSEMBLE AVERAGING)

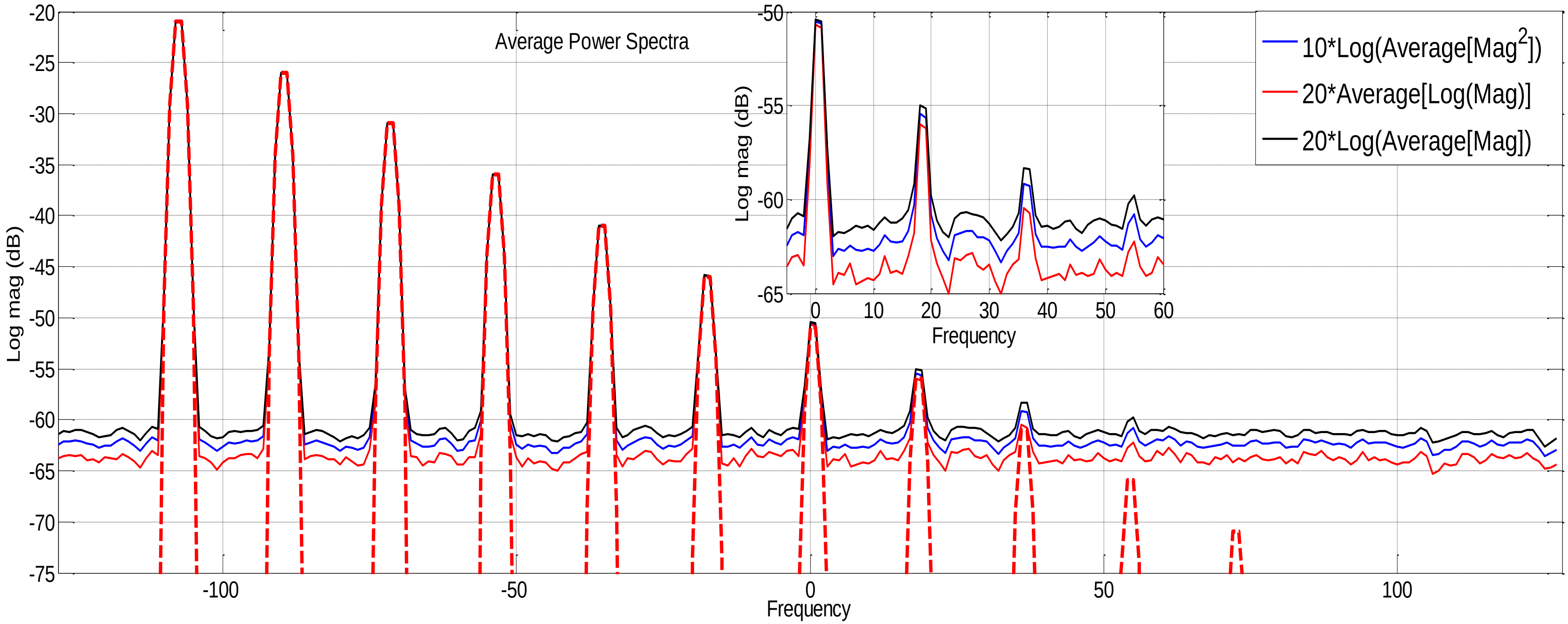
A Consistent Estimator:

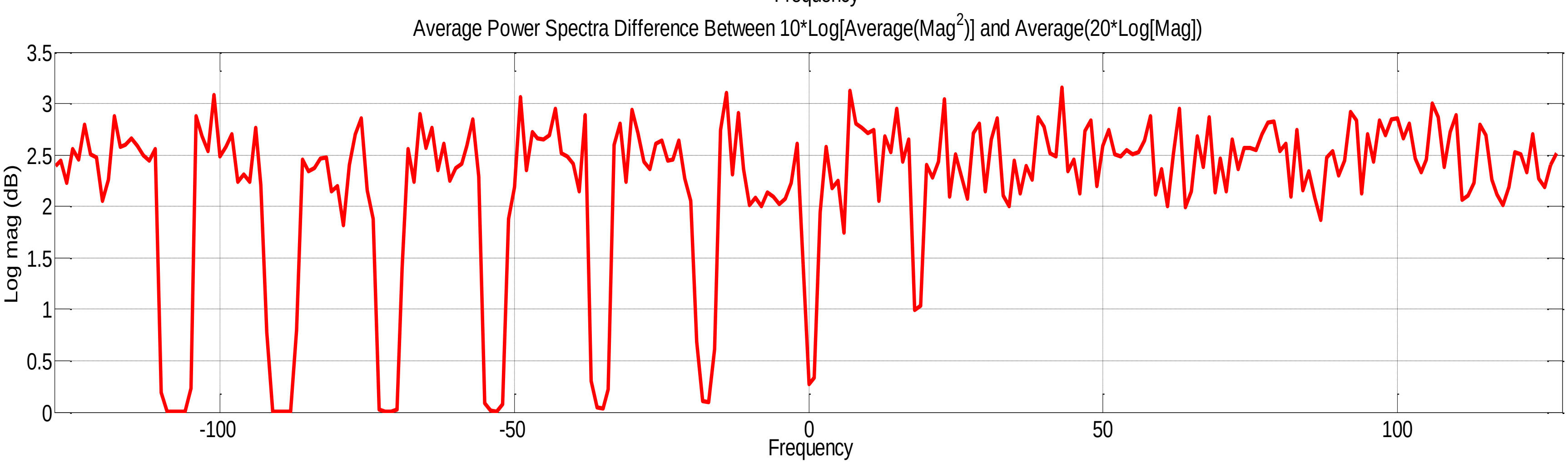
It would be nice if when you feed more data to an estimator, the quality of the estimate ~~improves~~ doesn't get worse!

Spectra: Ensemble Averaged FFTs of Sinewave and QAM Signal in White Noise

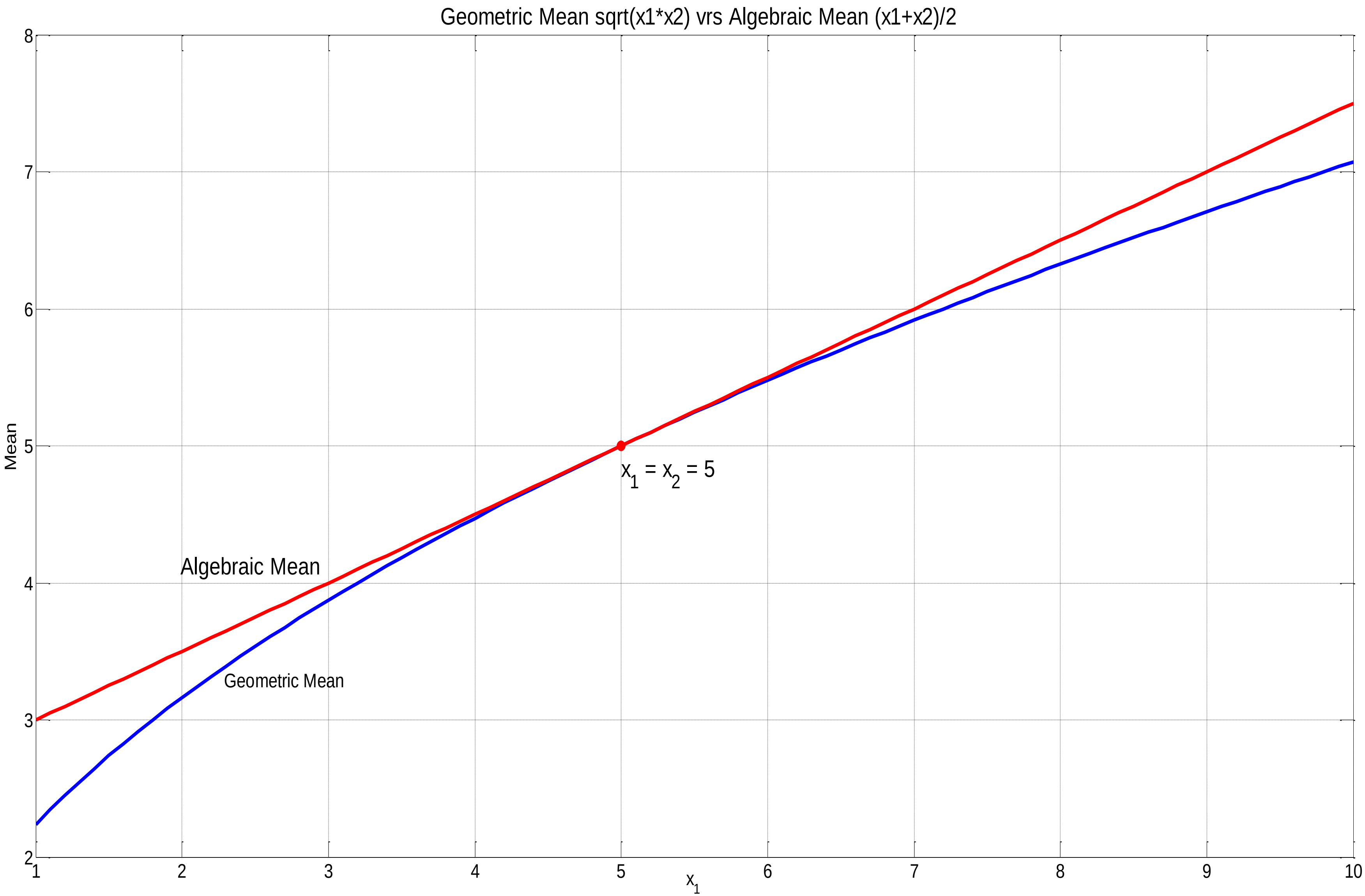


10 Log[Avg(mag^2)], 20 Log(Avg(Mag)], Avg[20 Log(Mag)]

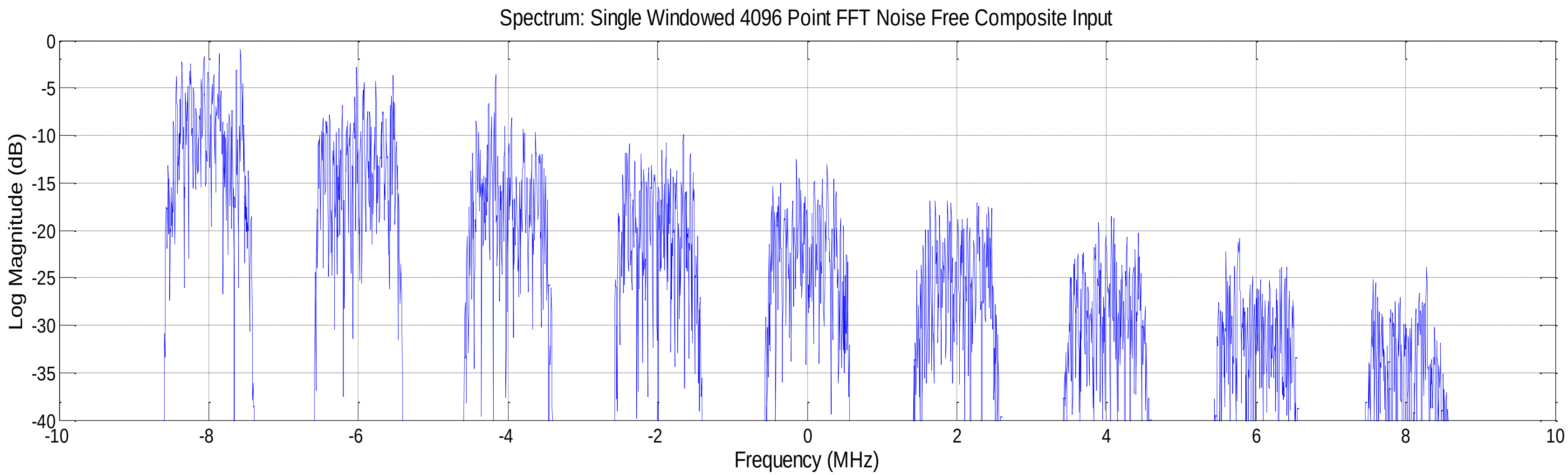


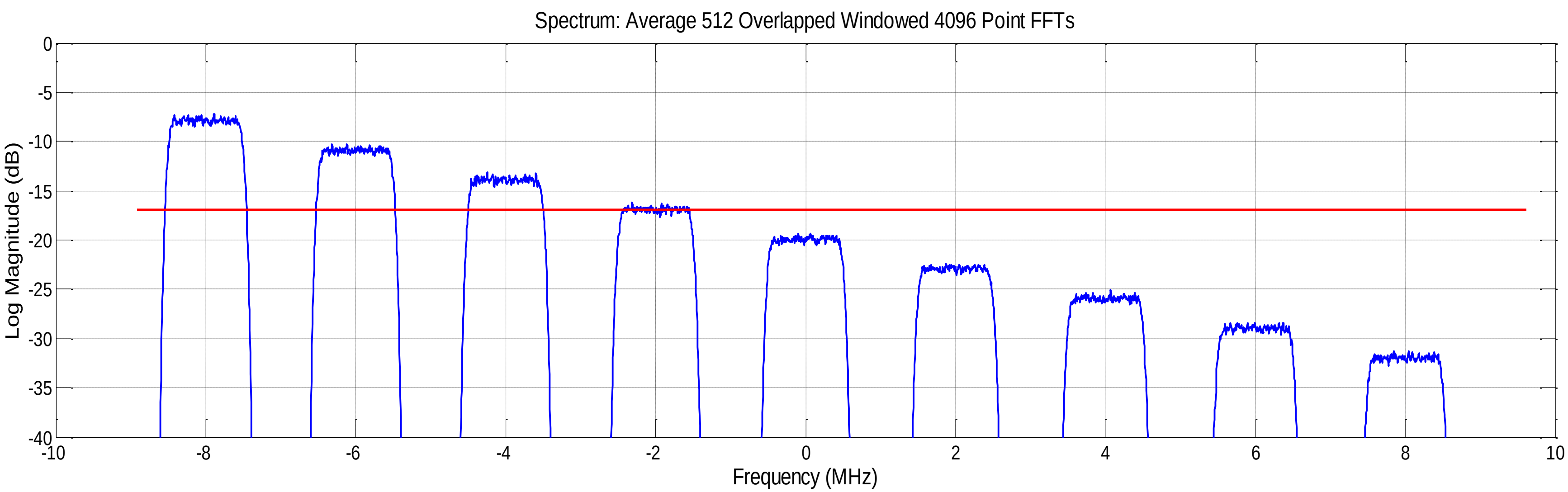


Geometric Mean Less Than or Equal to Algebraic Mean

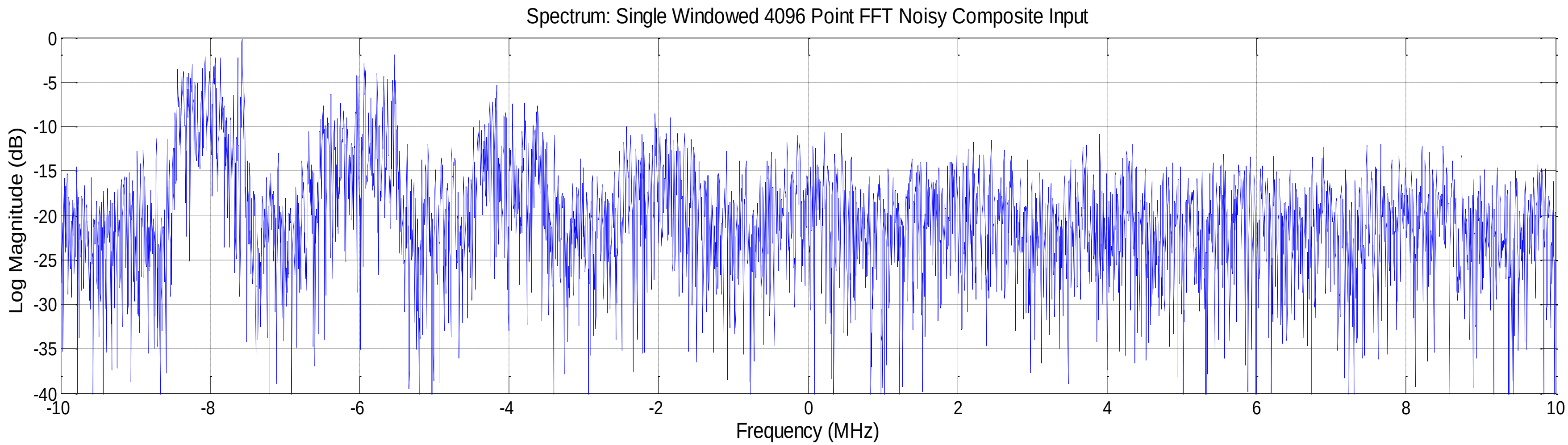


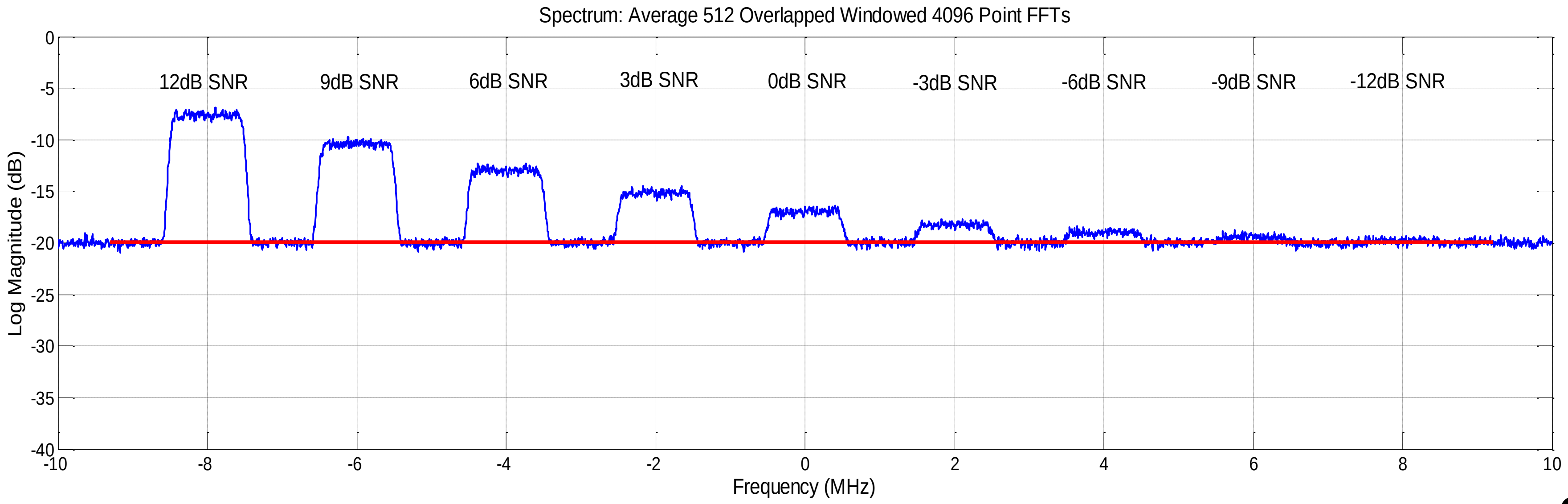
Raw Spectral Estimate and Averaged Spectral Estimate
Noise Free Signal 3-dB Steps in Average Power



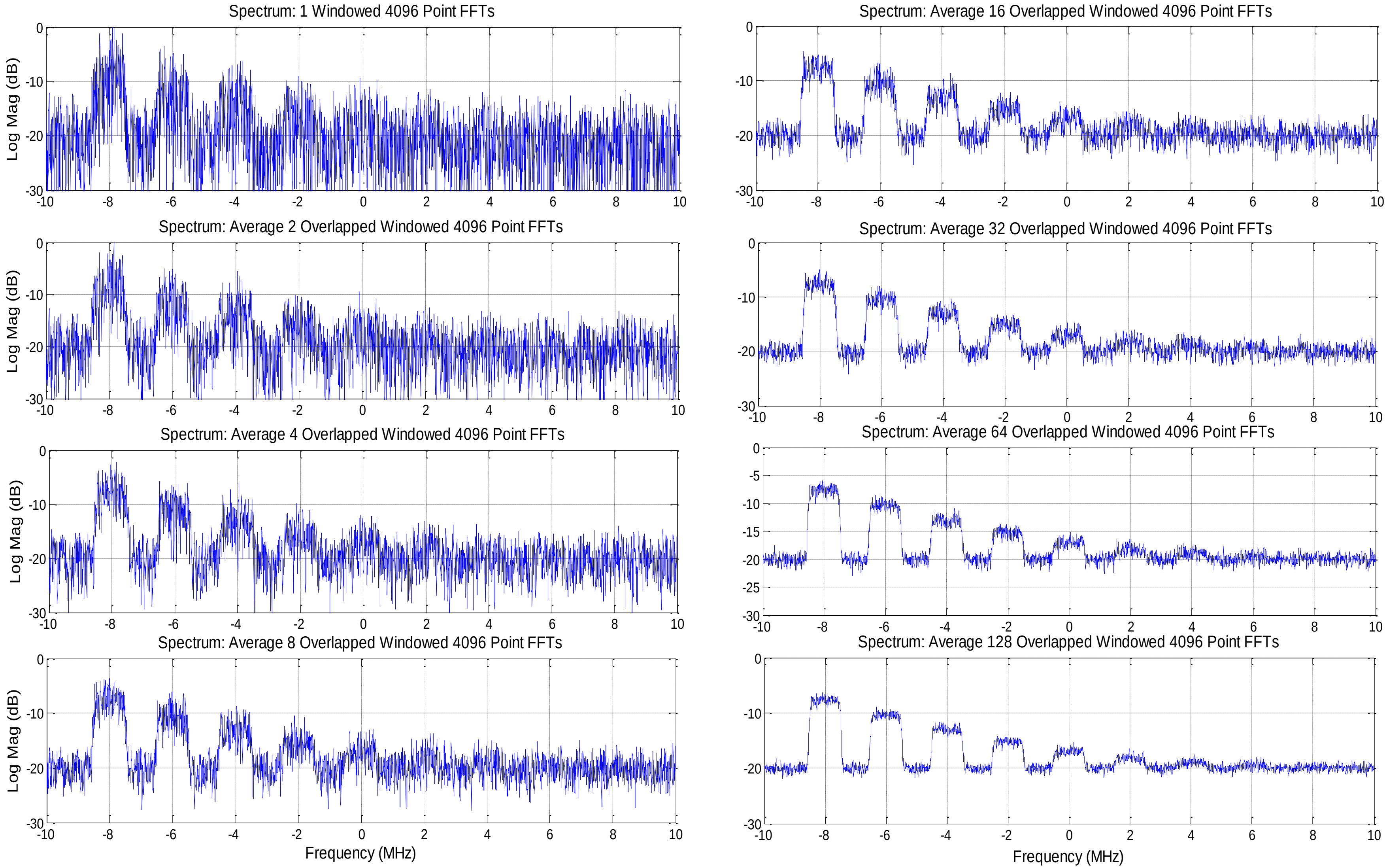


Raw Spectral Estimate and Averaged Spectral Estimate
Noise Plus Signal 3-dB Steps in Average Power

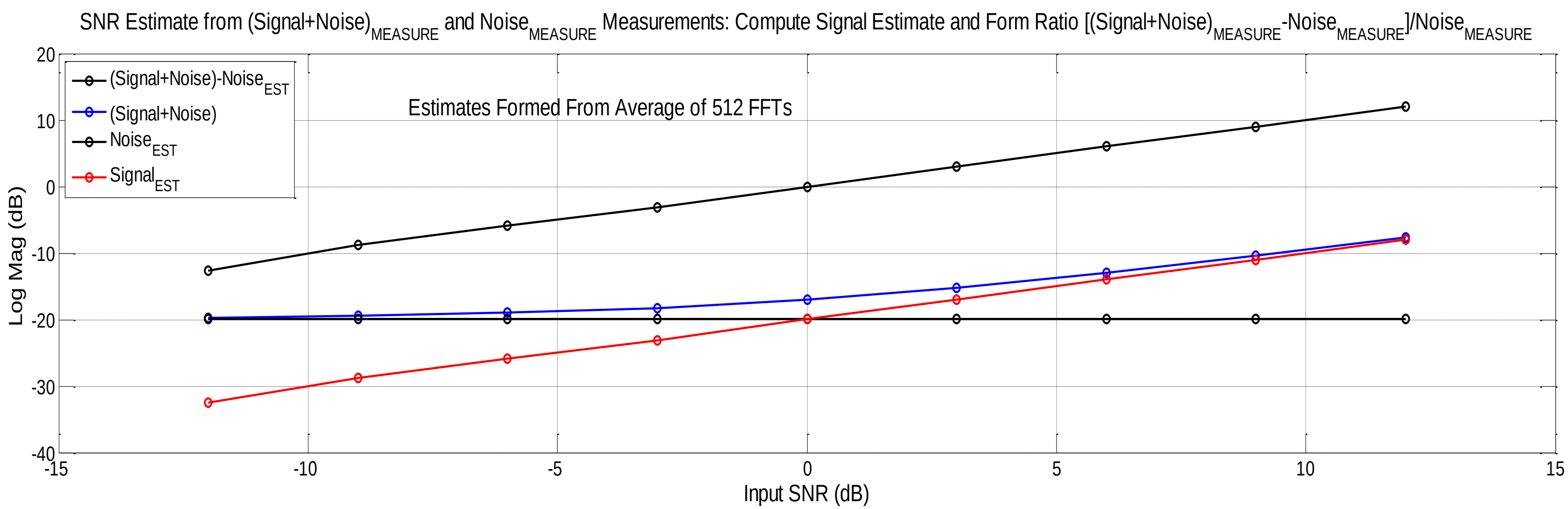


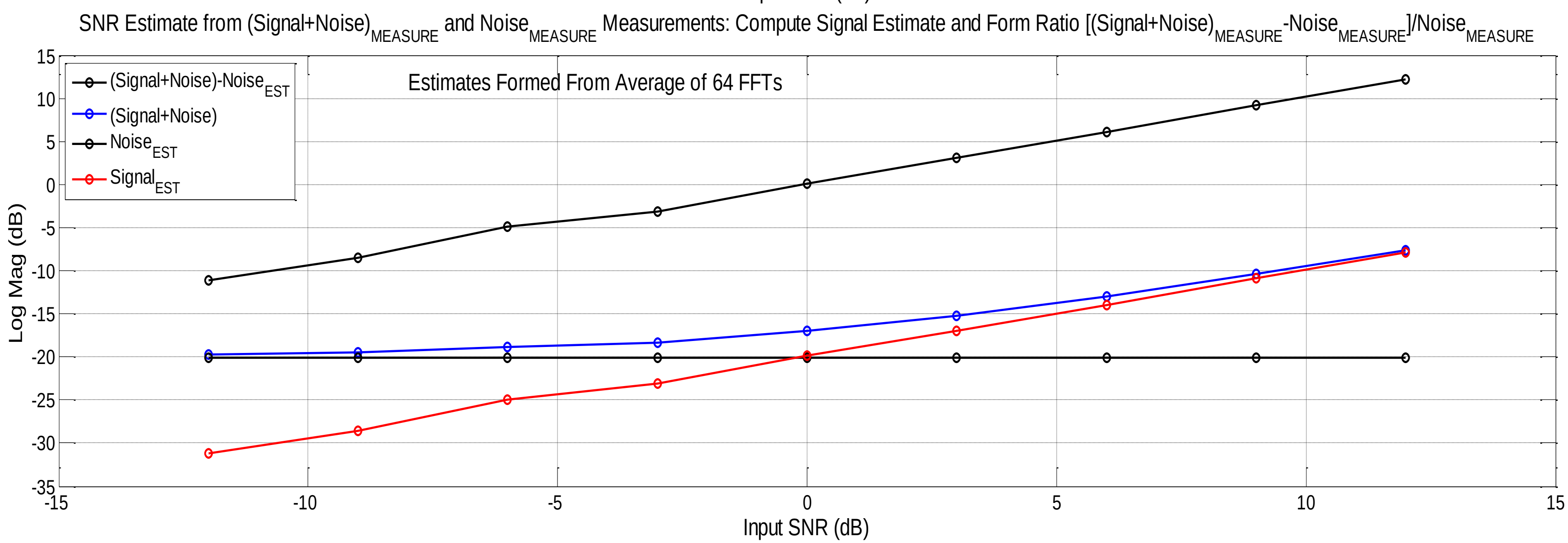


Variance Reduction for Successively Greater Number of Independent Terms in Ensemble Average



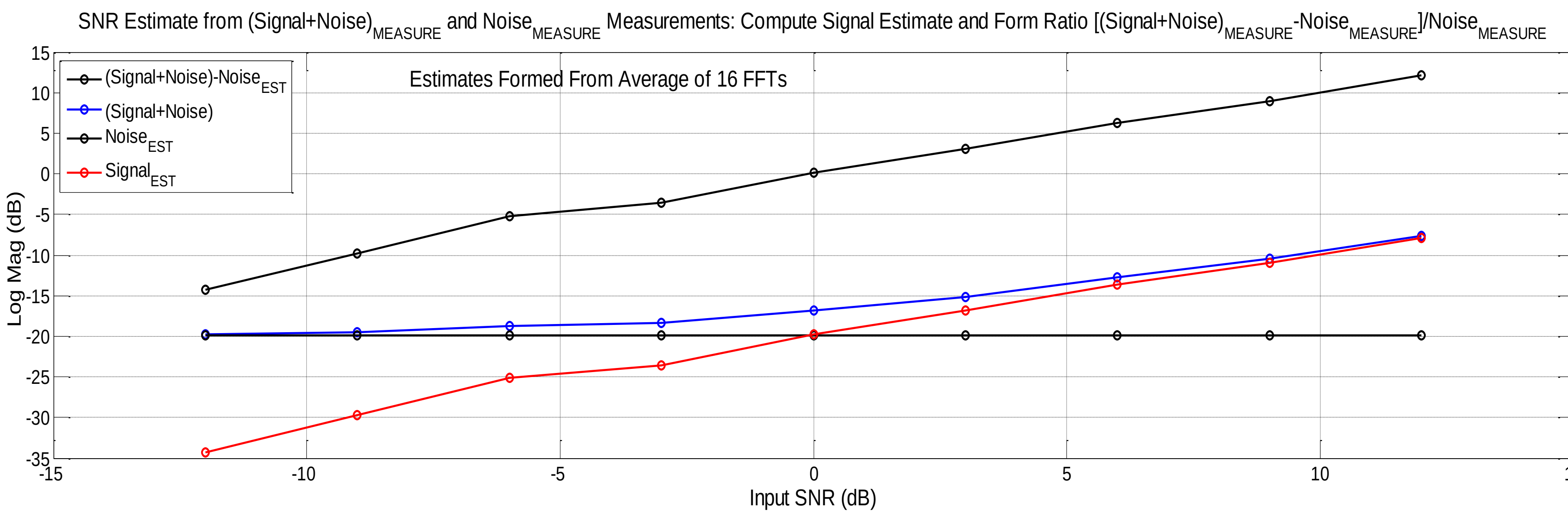
Estimate SNR From Measures of Signal Plus Noise and of Noise Only: SNR~((Signal+Noise)$_{Est}$-Noise$_{Est}$)/Noise$_{Est}$

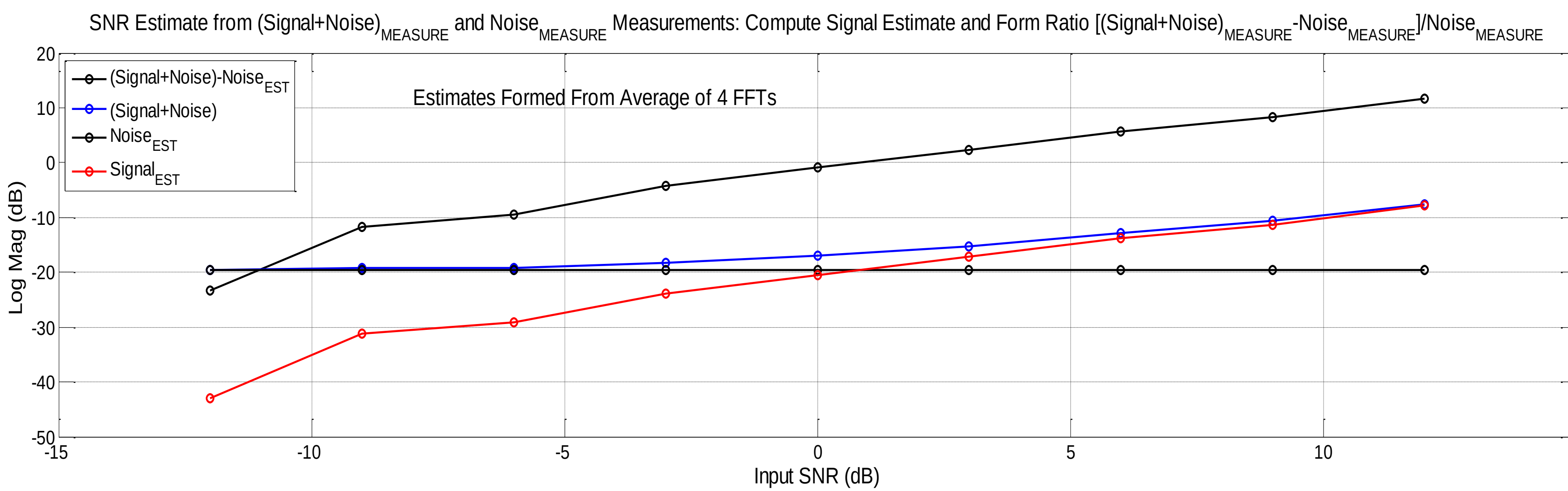




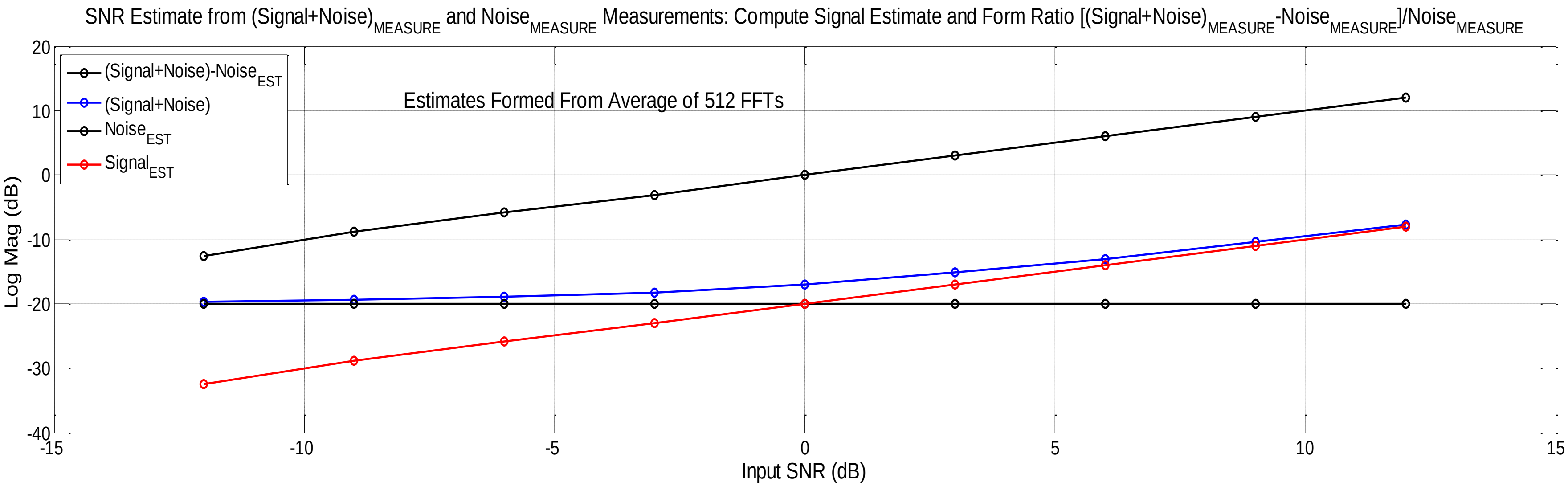
Estimate SNR From Measures of Signal Plus Noise and of Noise Only:

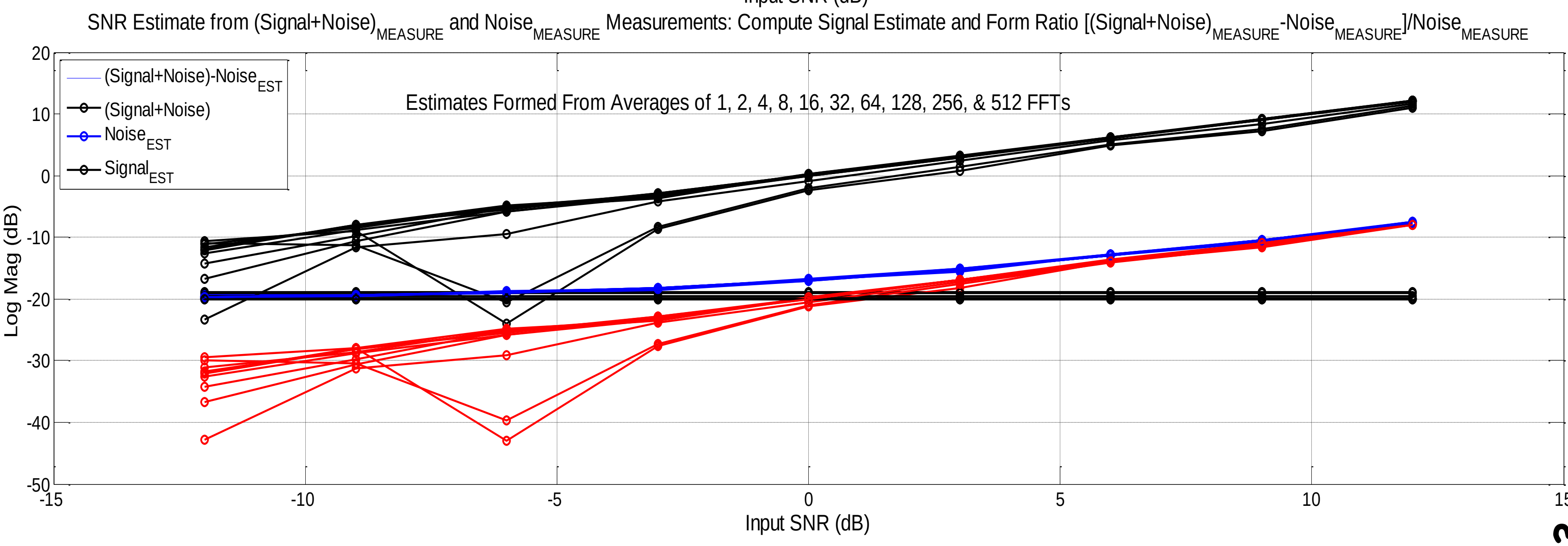
$SNR \sim ((Signal+Noise)_{Est} - Noise_{Est}) / Noise_{Est}$



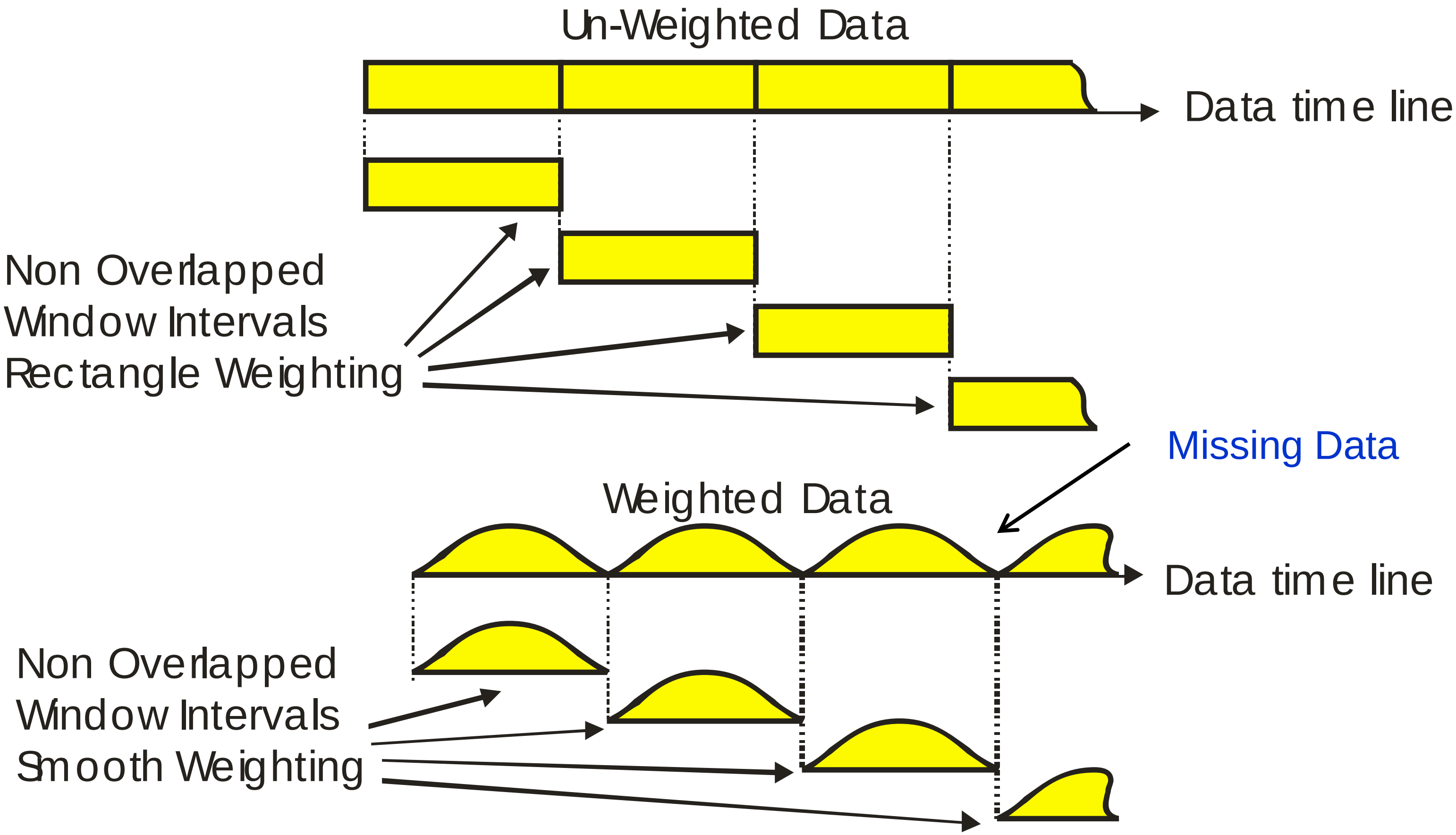


Estimate SNR From Measures of Signal Plus Noise and of Noise Only: SNR~((Signal+Noise)$_{Est}$-Noise$_{Est}$)/Noise$_{Est}$

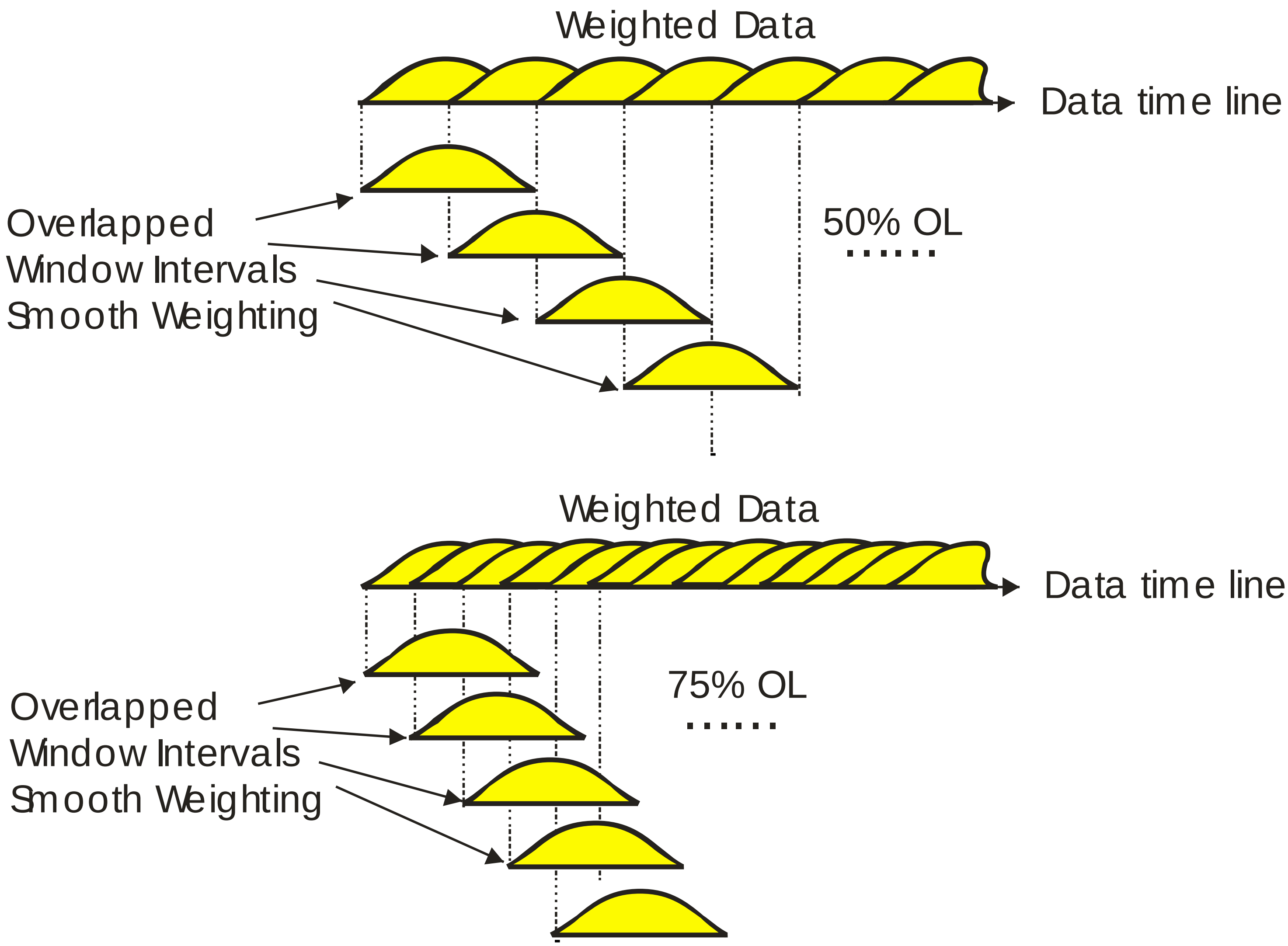




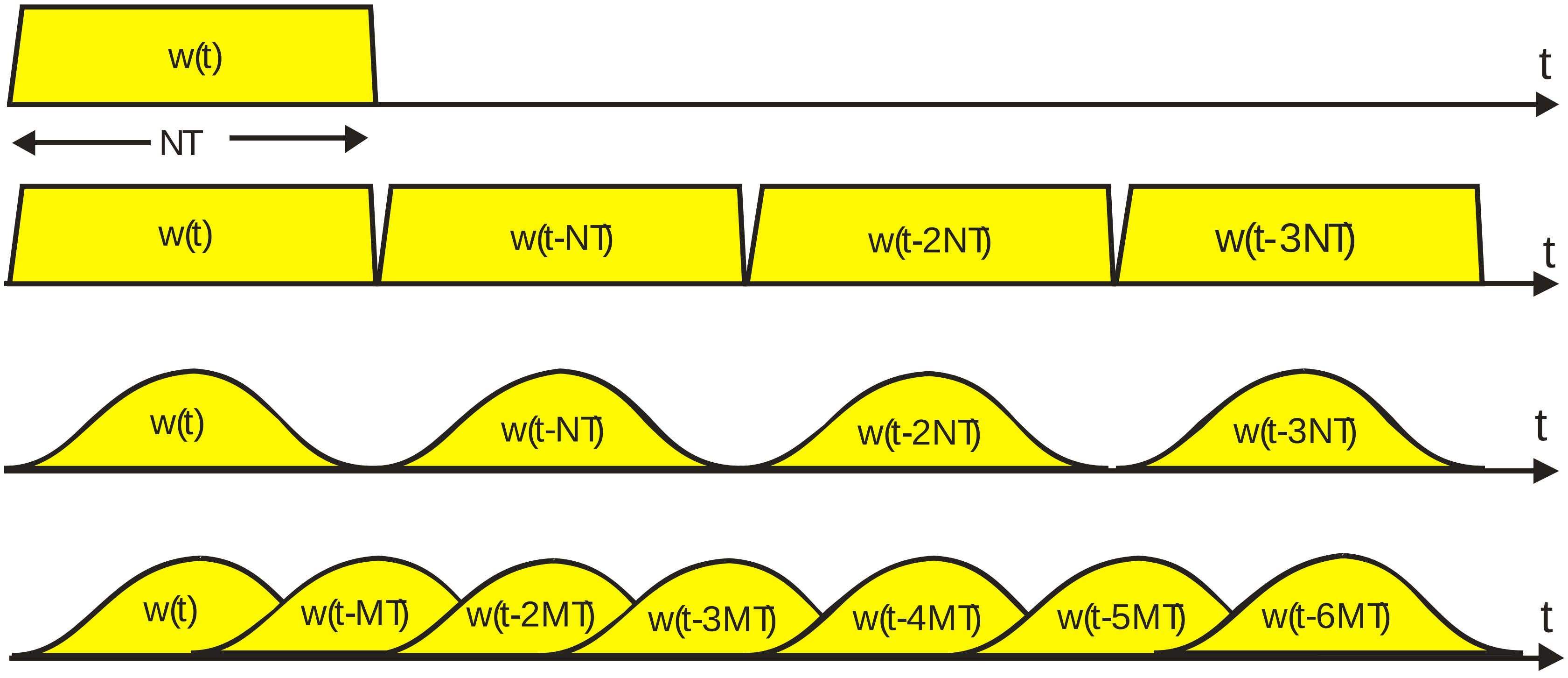
Window to Suppress Boundary Effects



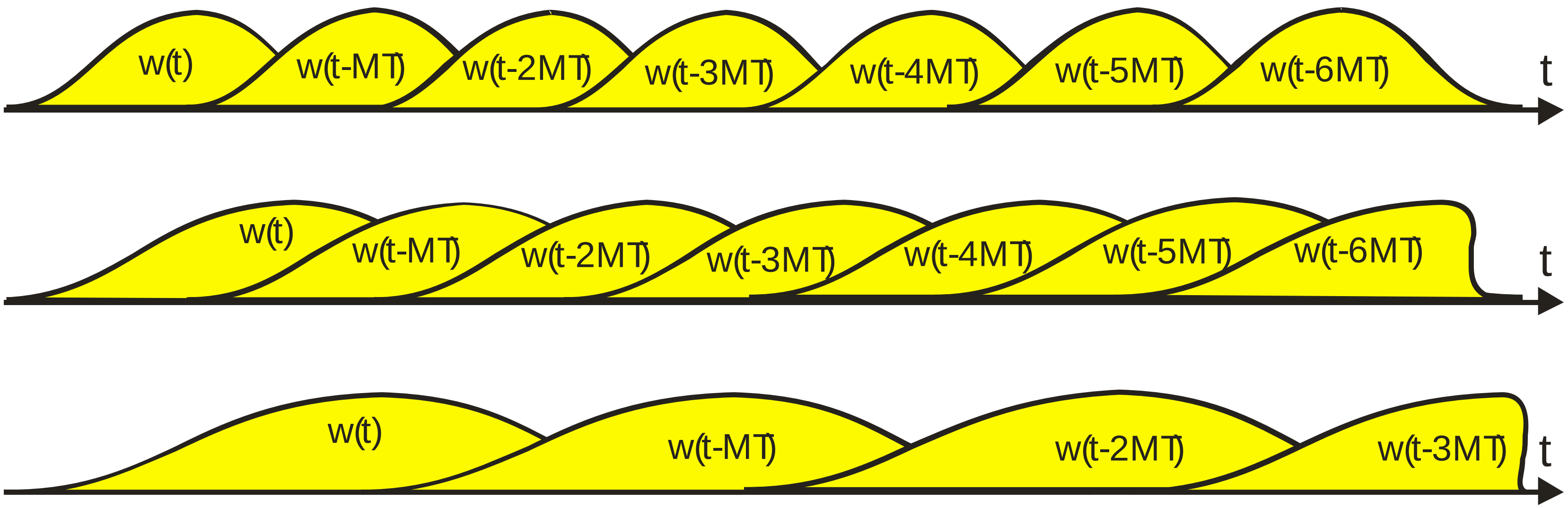
How Much Overlap?



SLIDING, OVERLAPPED, AND WINDOWED FFTS

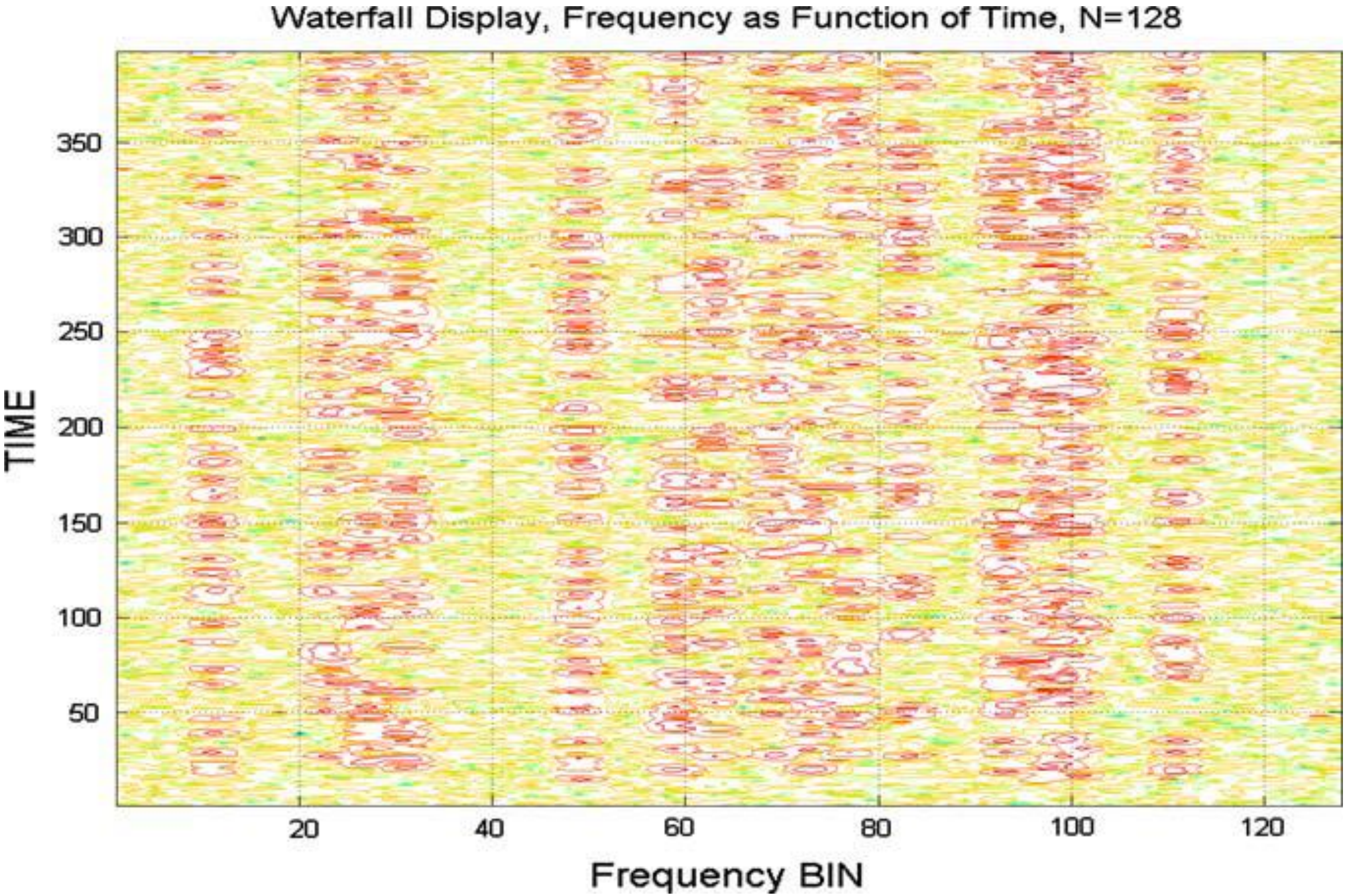


WINDOW LENGTH AND OVERLAP

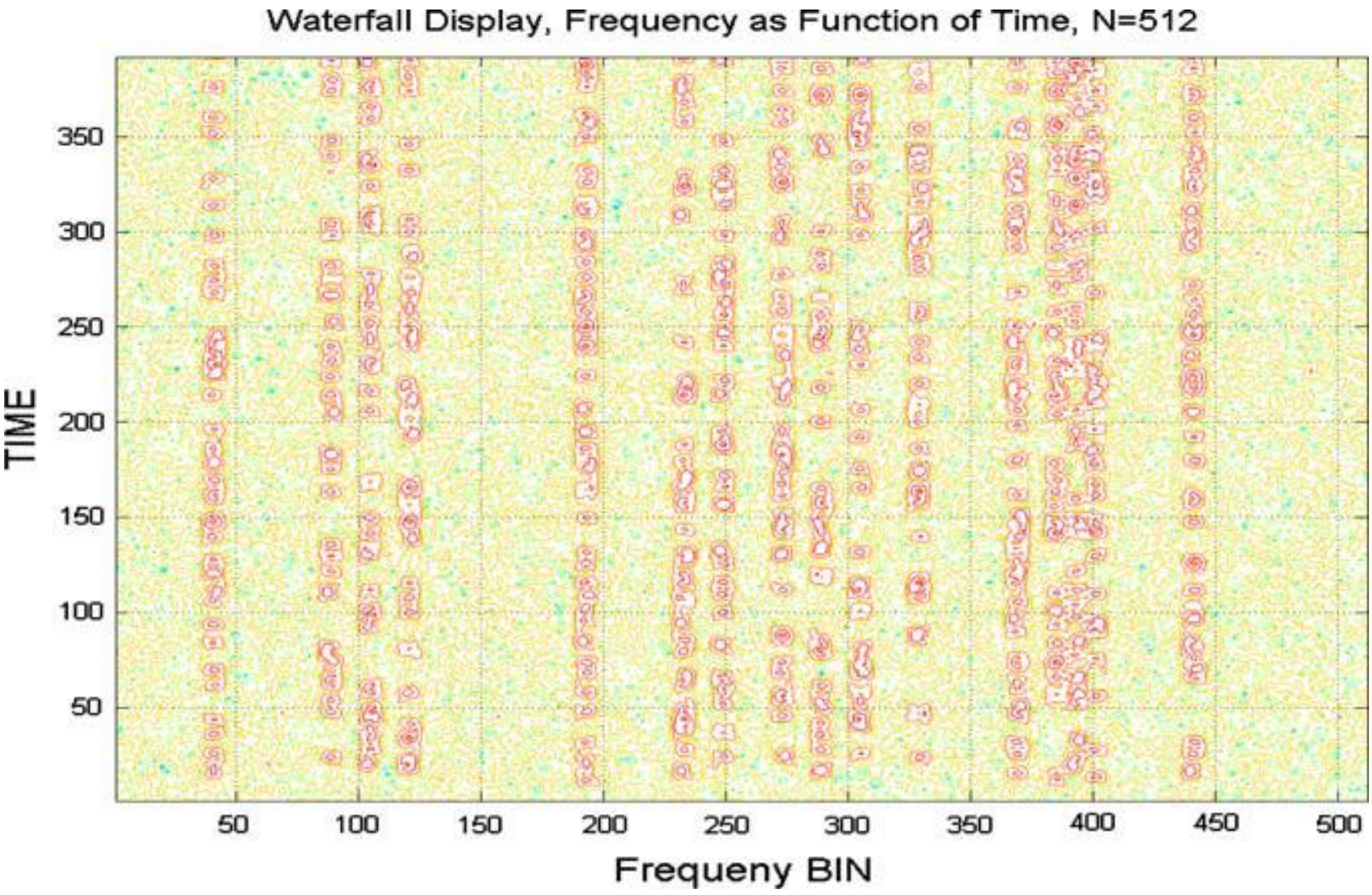


Short Windows, Short Averaging Time,
Quick Response to Change, Poor Frequency Resolution

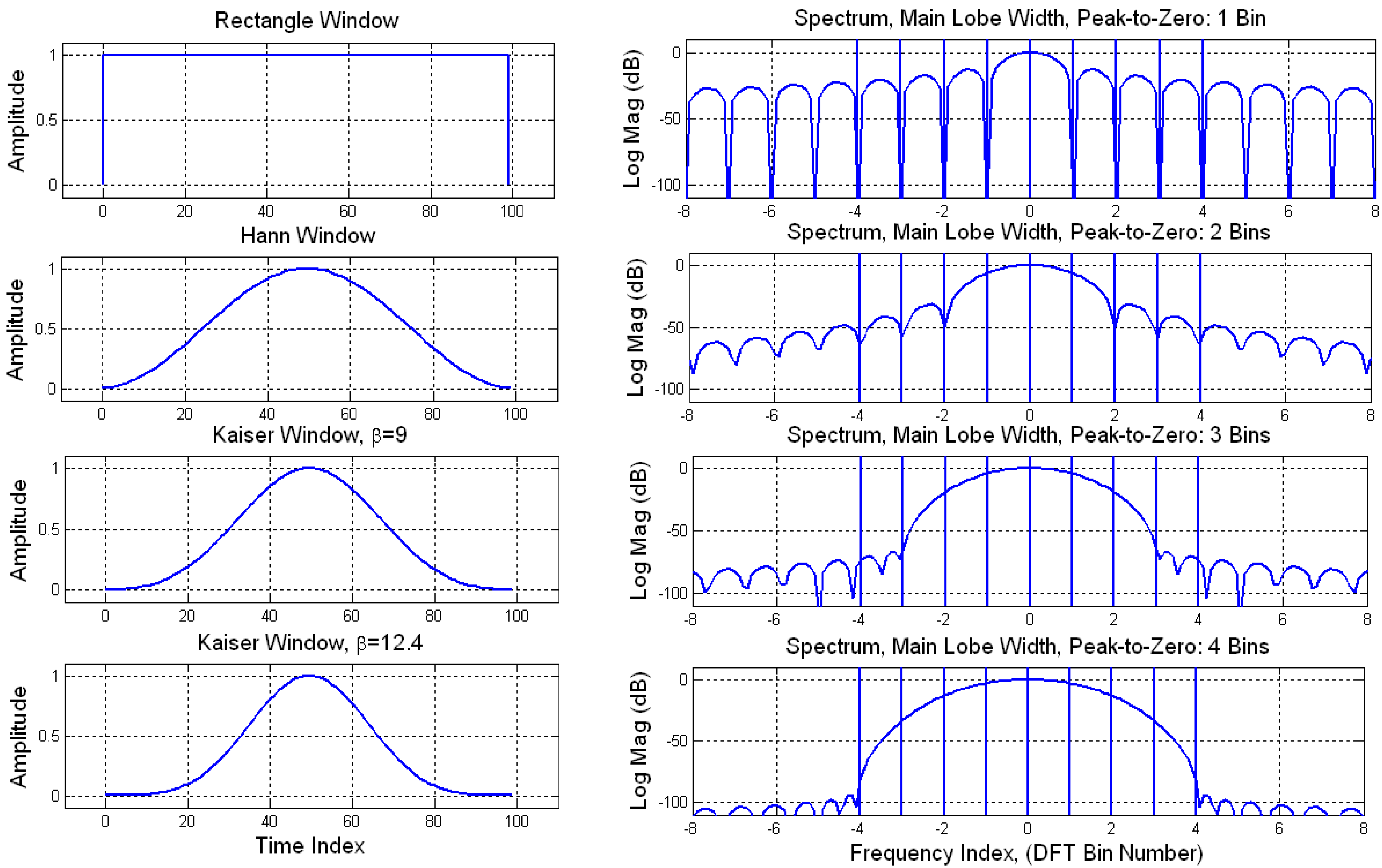
Longer Windows, Longer Averaging Time,
Slow Response to Change, Good Frequency Resolution



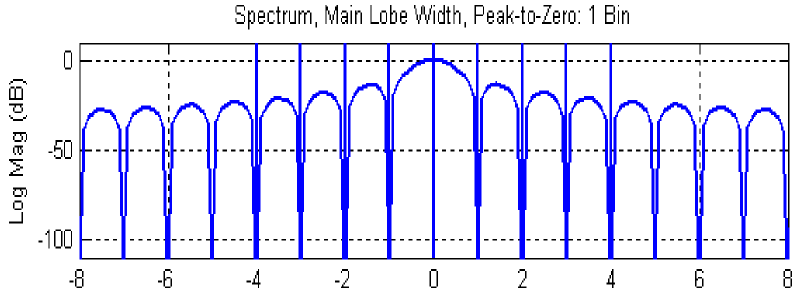
Waterfall Display, Frequency as Function of Time, N=128
TIME
350
300
250
200
150
100
50
20
40
60
80
100
120
Frequency BIN


Waterfall Display, Frequency as Function of Time, N=512
TIME
350
300
250
200
150
100
50
50
100
150
200
250
300
350
400
450
500
Frequeny BIN

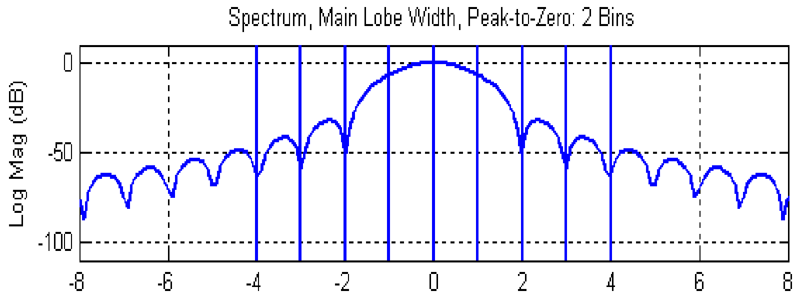
WINDOWS
TRADE INCREASED SPECTRAL MAIN LOBE WIDTH FOR REDUCED LEVEL SPECTRAL SIDE LOBES



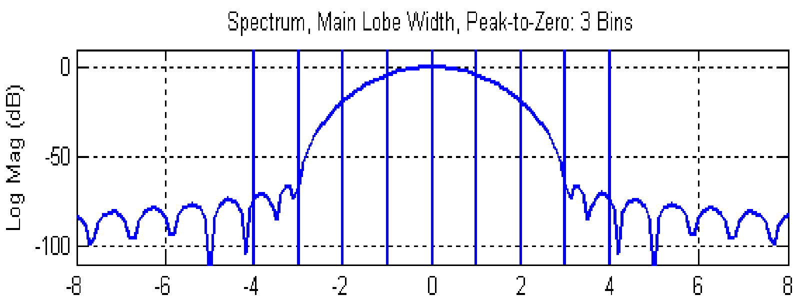
Satisfy Nyquist Criterion, Examine Main Lobe BW



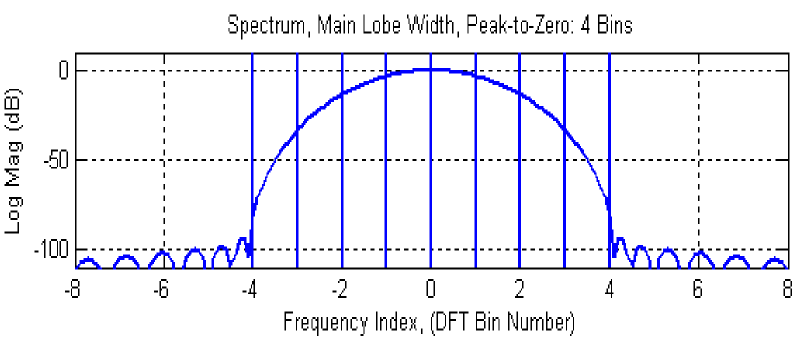
Main Lobe BW = fs/N
Nyquist Rate = fs/N (No Overlap)
Perform N-to-1 Down Sample in DFT
Input N-samples, Output 1 Output Sample



Main Lobe BW = 2fs/N fs/N
Nyquist Rate = 2fs/N = fs/(N/2) (50% Overlap)
Perform N-to-2 Down Sample in DFT
Input N/2-samples, Output 1 Output Sample

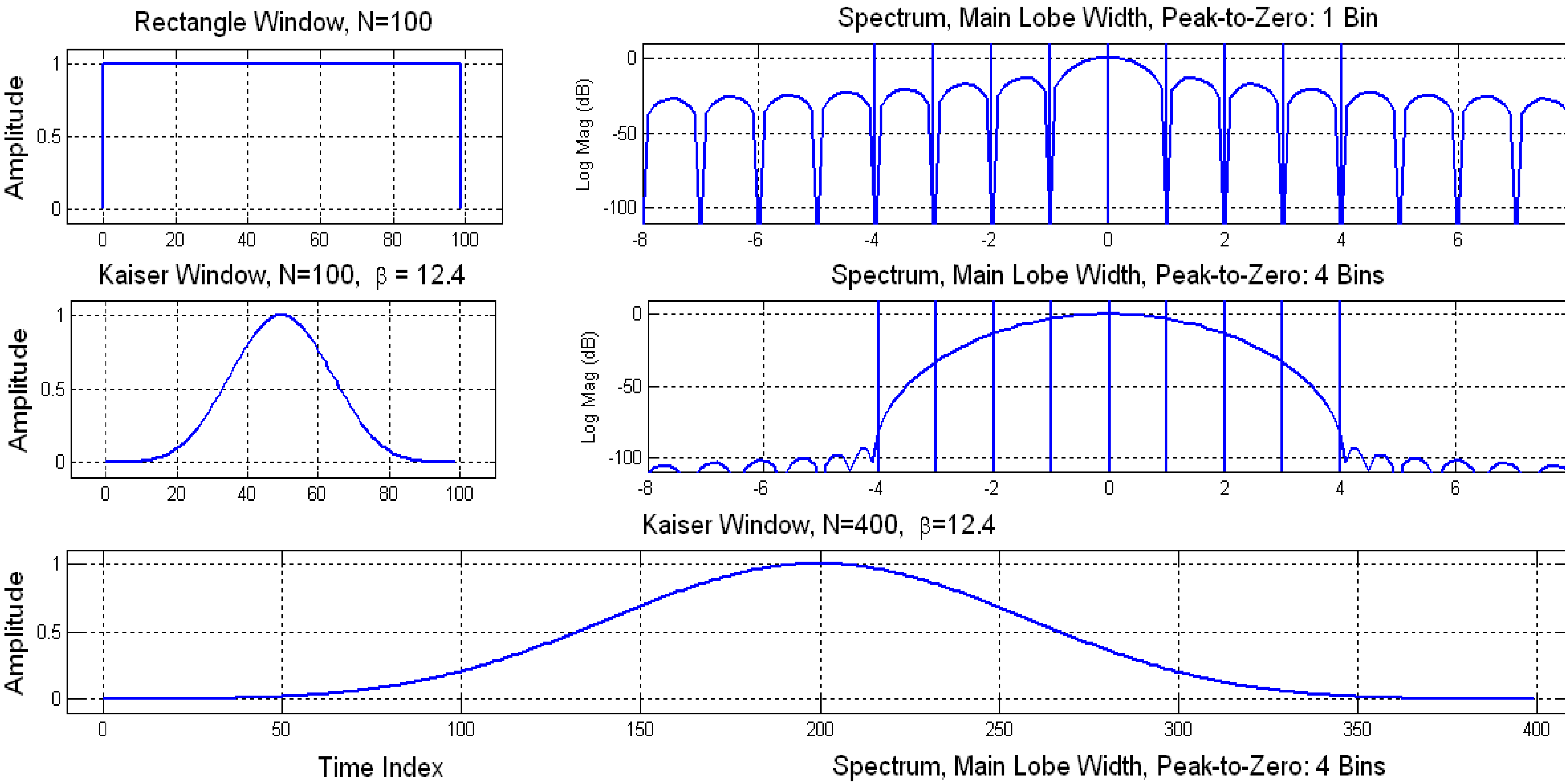


Main Lobe BW = 3fs/N fs/N
Nyquist Rate = 3fs/N = fs/(N/3) (66.6% Overlap)
Perform N-to-3 Down Sample in DFT
Input N/3-samples, Output 1 Output Sample

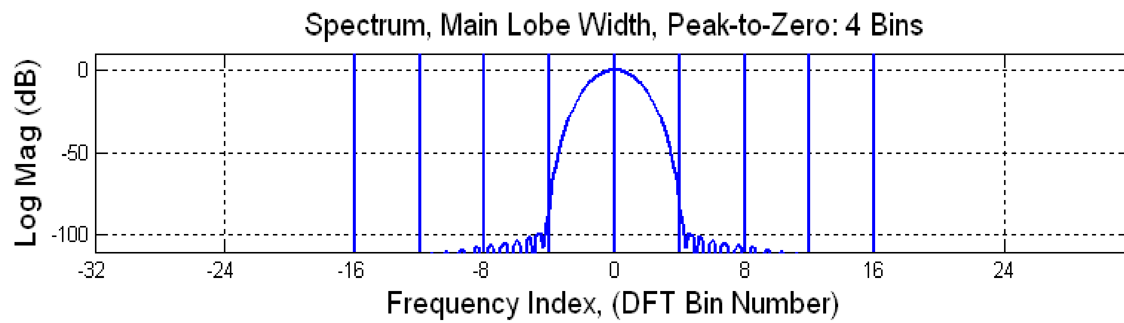


Main Lobe BW = 4fs/N
Nyquist Rate = 4fs/N = fs/(N/4) (75% Overlap)
Perform N-to-4 Down Sample in DFT
Input N/4-samples, Output 1 Output Sample

N POINT SMOOTH WINDOW: BW IS 4-TIMES WIDTH OF RECTANGLE WINDOW
4N POINT SMOOTH WINDOW: BW IS EQUAL TO WIDTH OF RECTANGLE WINDOW

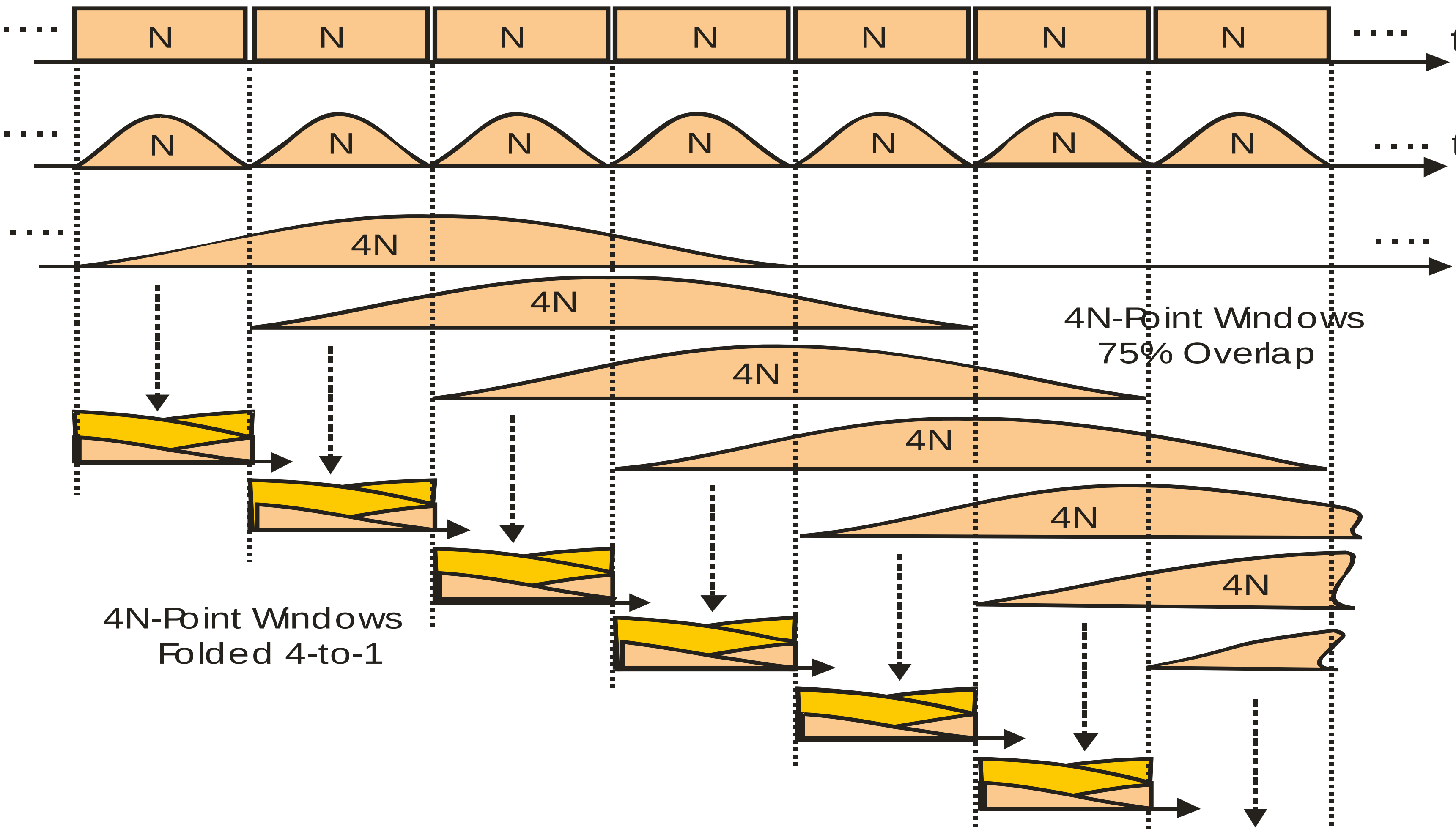


Bin 4 of 4N Point FFT
is same BW as
Bin 1 of N Point FFT

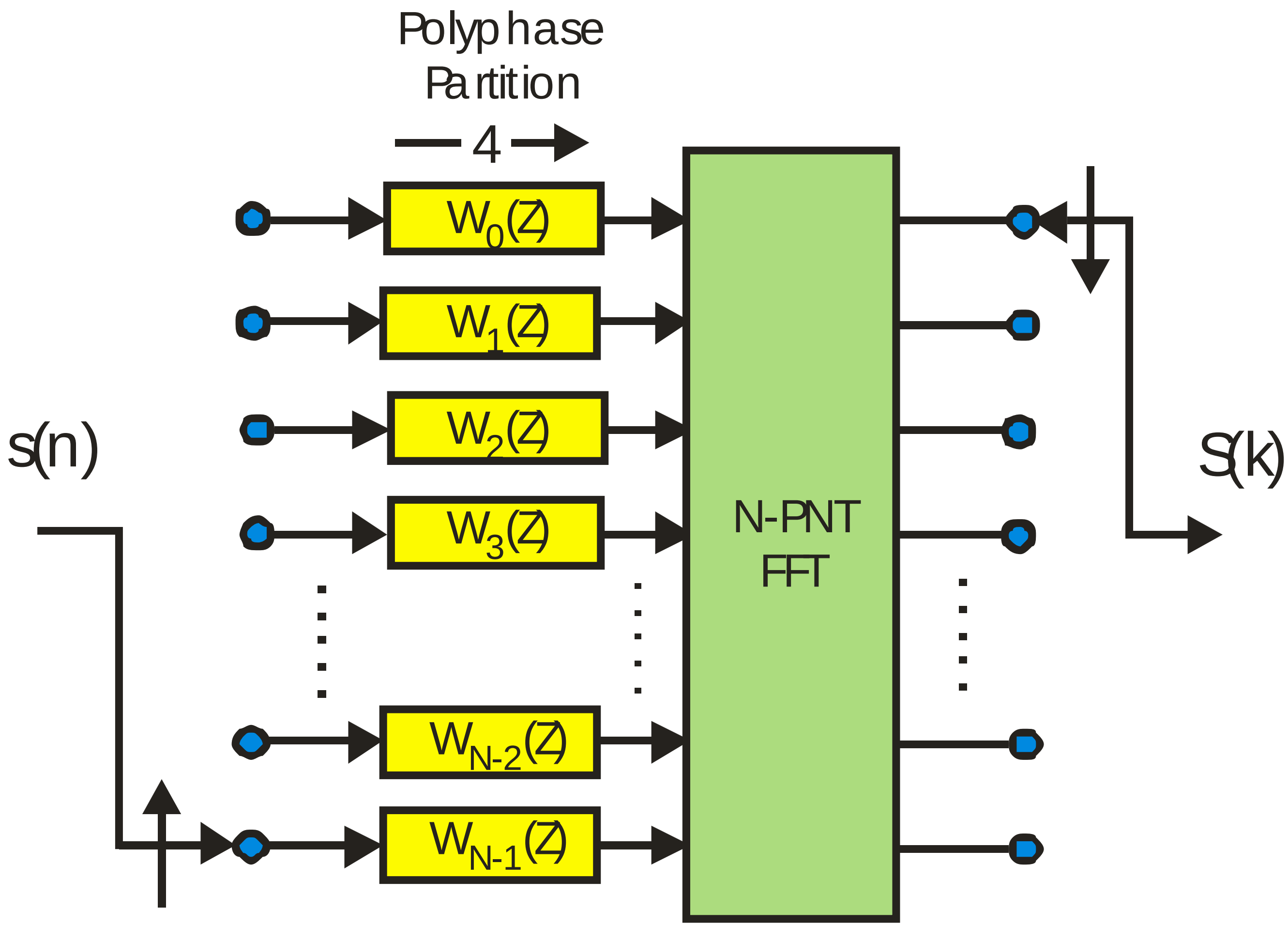


SUCCESSIVE NON OVERLAPPED WINDOW INTERVALS OF LENGTH N
OVERLAPPED WINDOW INTERVALS OF LENGTH 4N,
AND 4-TO-1 FOLDED (OR ALIASED) WINDOW INTERVALS

FOLDED WINDOW IS A POLYPHASE PARTITION



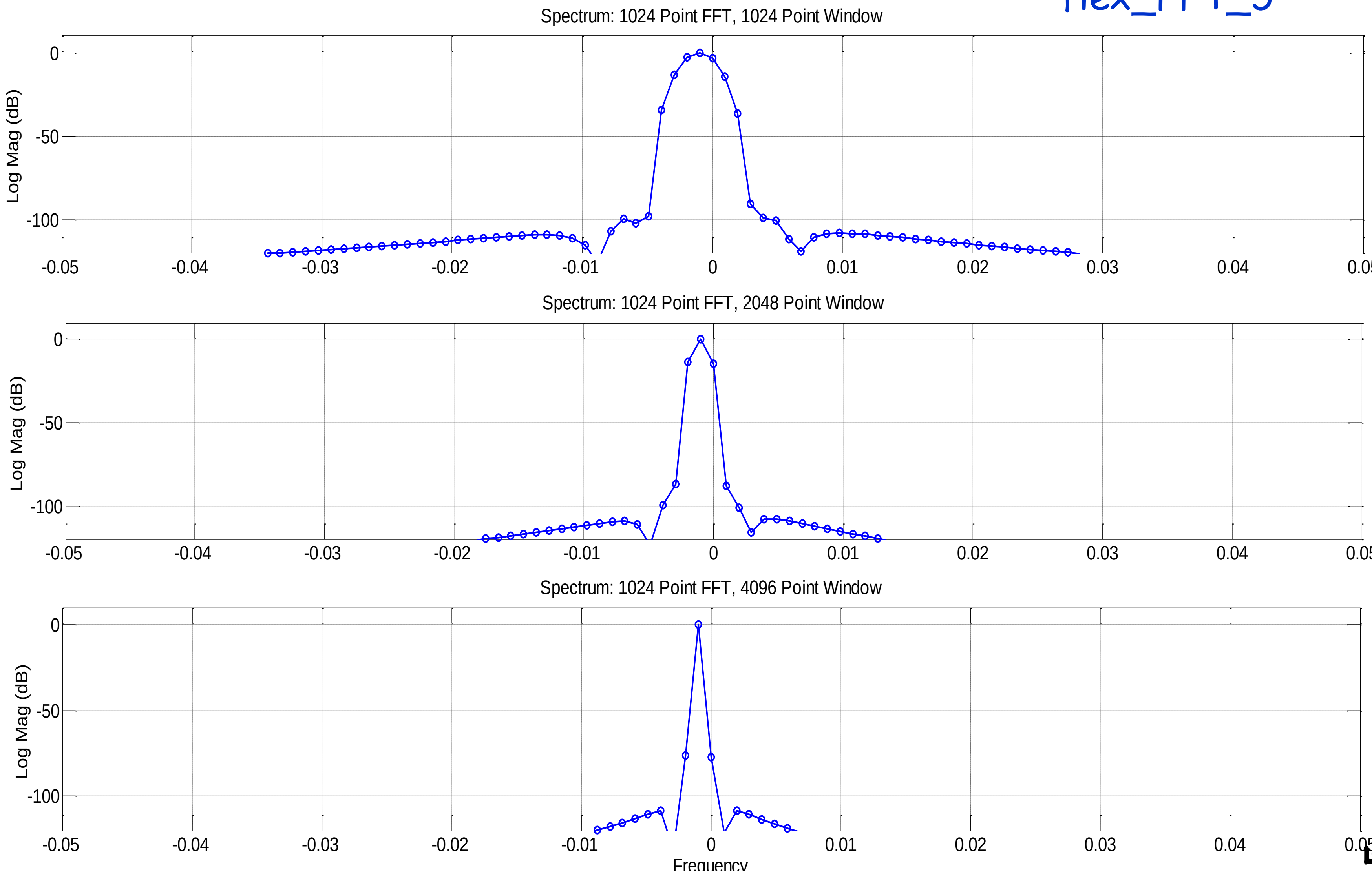
4-N POINT WINDOW FOLDED 4-TO-1 BY POLY-PHASE PARTITION AS INPUT TO N-POINT FFT



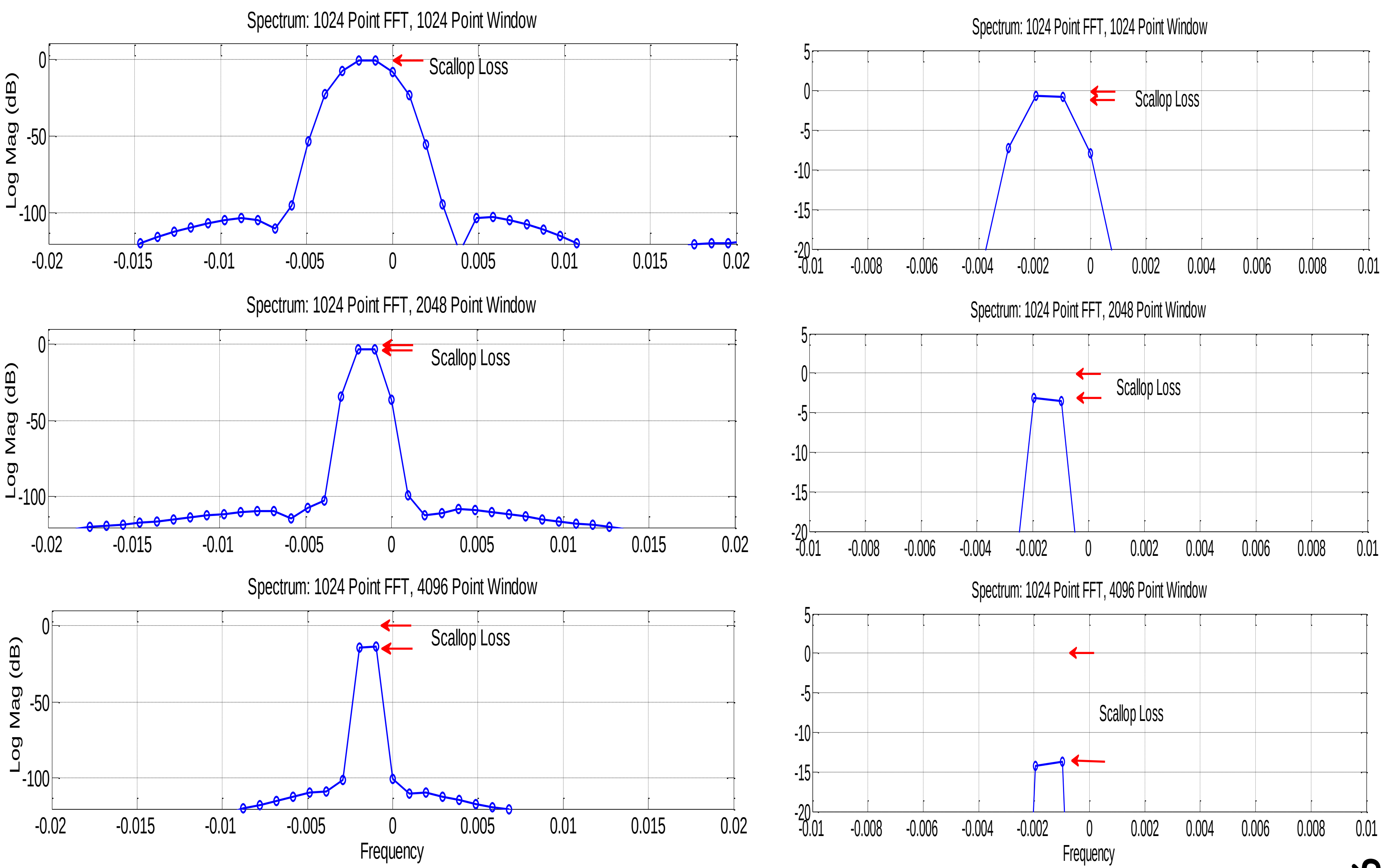
$$W_r(Z) = Z\{w(r+nN)\}$$

Frequency Response of 1024 Point FFT with 1024 Point Window, 2048 Point Window Folded 2-to-1, and 4096 Point Window Folded 4-to-1

flex_FFT_5

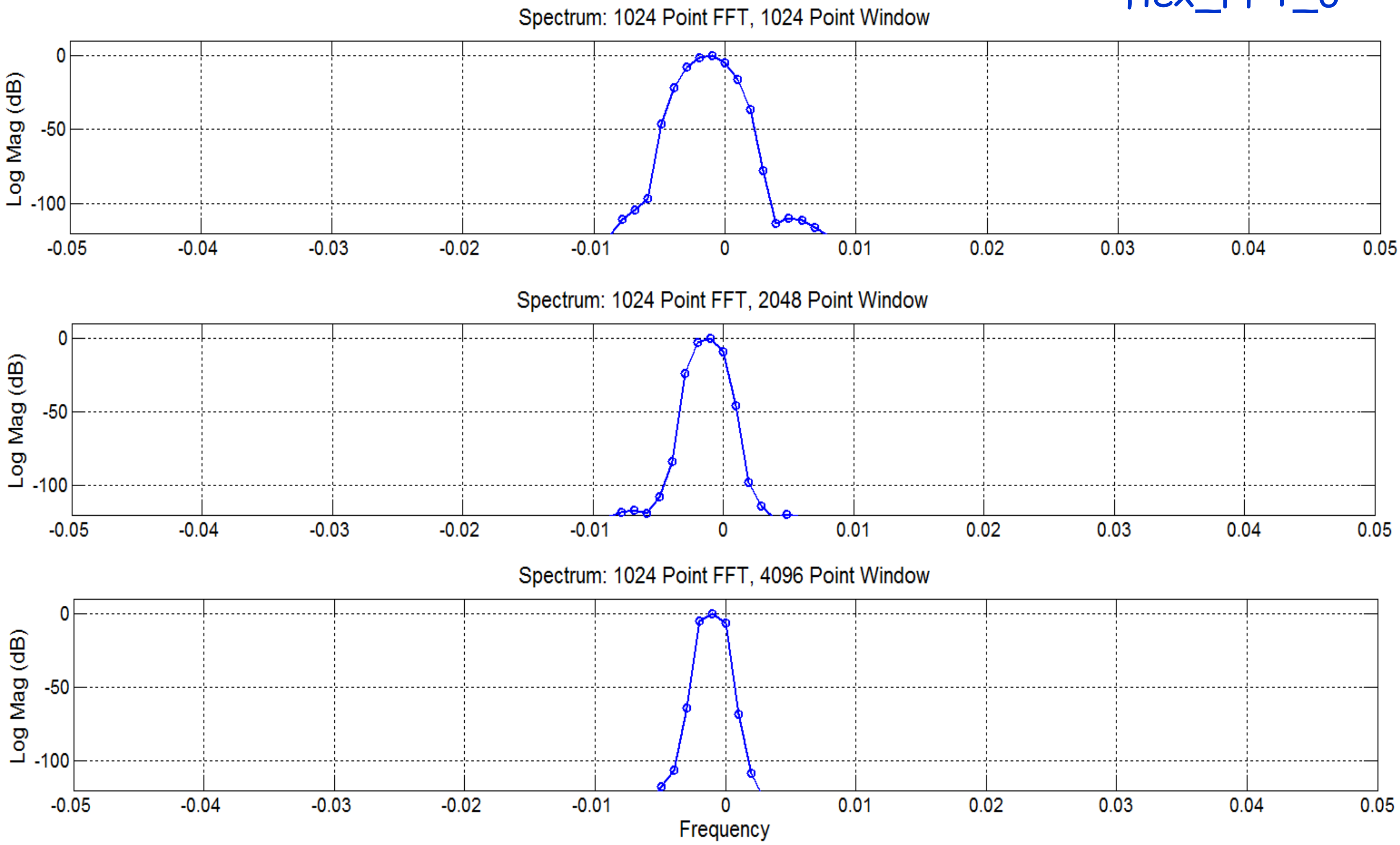


Scallop Loss For Sinusoid Displaced Half Bin Width

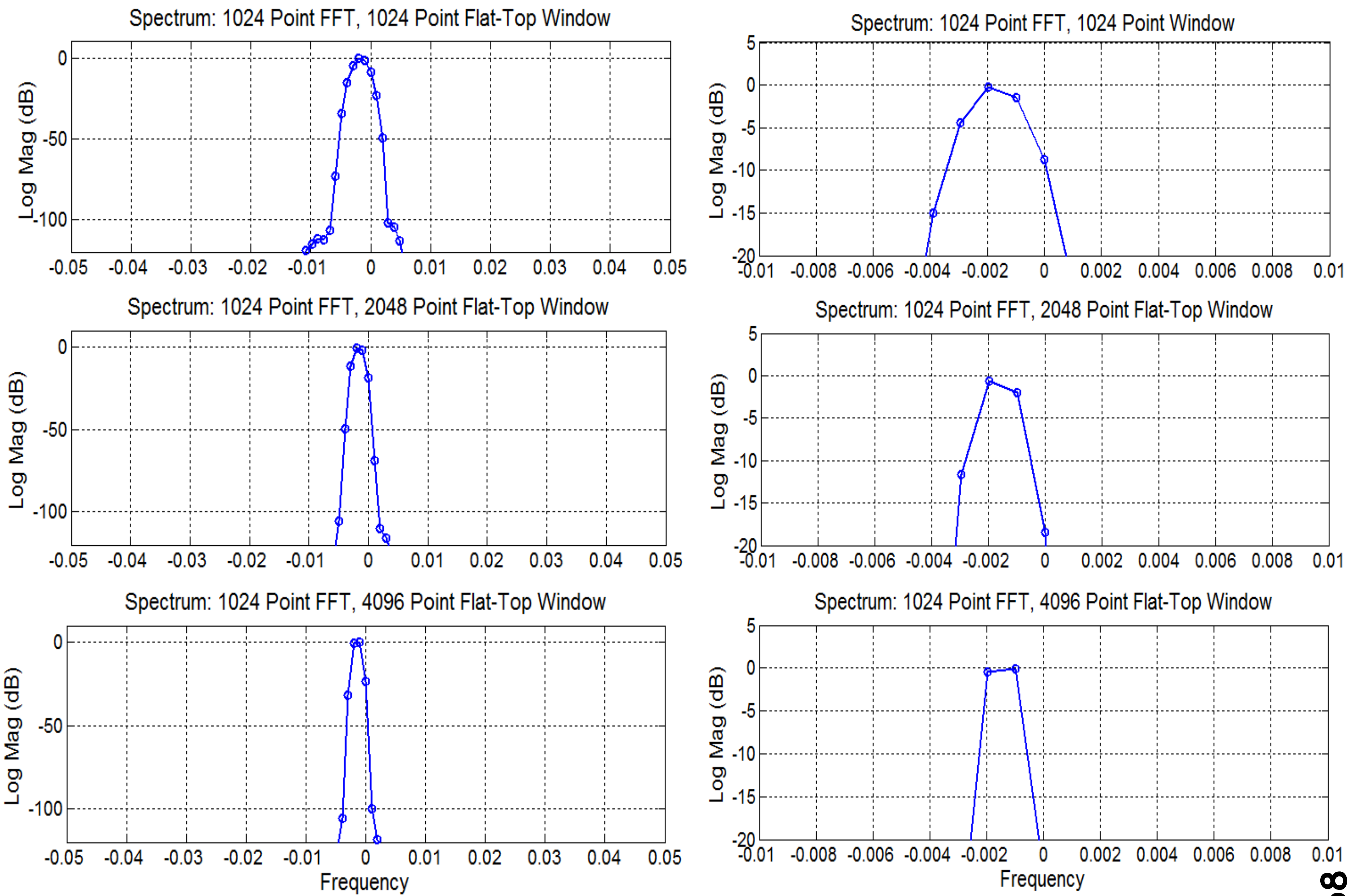


Frequency Response of 1024 Point FFT with 1024 Point Window, 2048 Point Window Folded 2-to-1, and 4096 Point Window Folded 4-to-1, (Flat-Top Windows)

flex_FFT_6

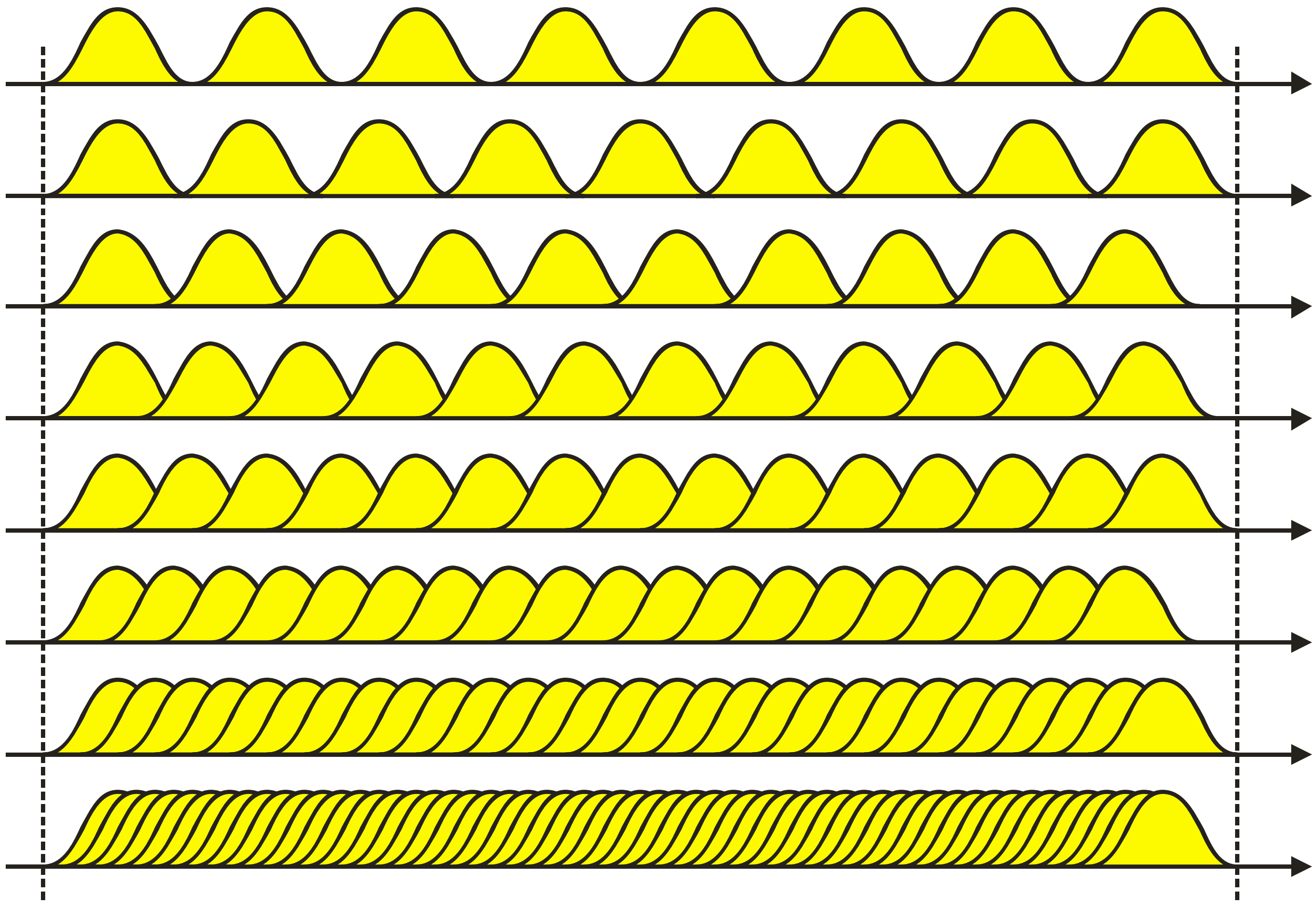


Flat Top Scallop Loss For Sinusoid Displaced Half Bin Width

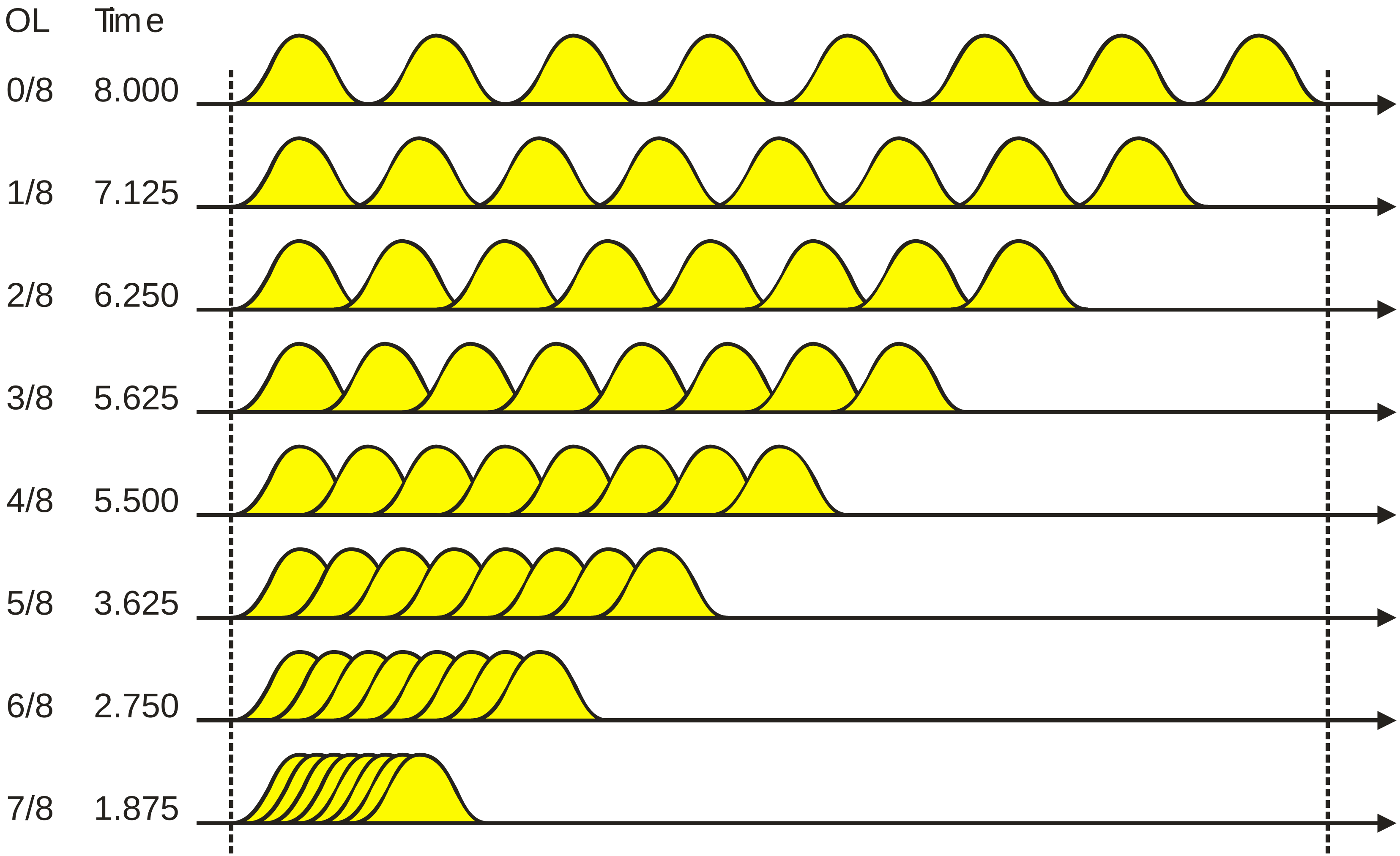


OVERLAPPED WINDOWS: MORE OVERLAP INCREASED NUMBER OF WINDOWS PER FIXED INTERVAL, INCREASED CORRELATION IN SUCCESSIVE WINDOWS

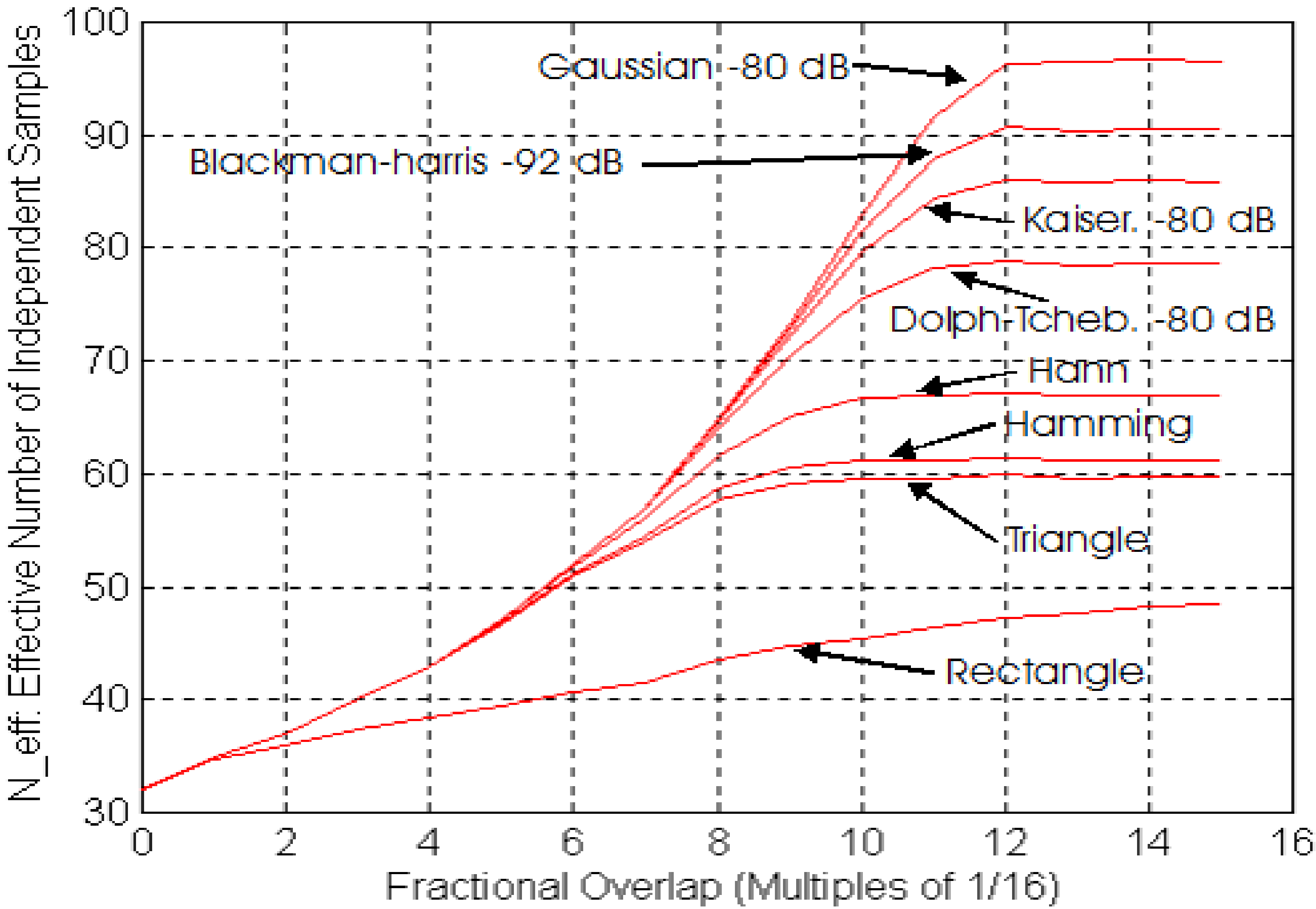
OL	No.	Corr
0/8	8	0.0000
1/8	9	0.0002
2/8	10	0.0034
3/8	12	0.0255
4/8	15	0.1069
5/8	19	0.2963
6/8	29	0.5892
7/8	57	0.8776



OVERLAPPED WINDOWS: MORE OVERLAP DECREASES TIME INTERVAL REQUIRED TO ACQUIRE SPECIFIED NUMBER OF WINDOWED INTERVALS

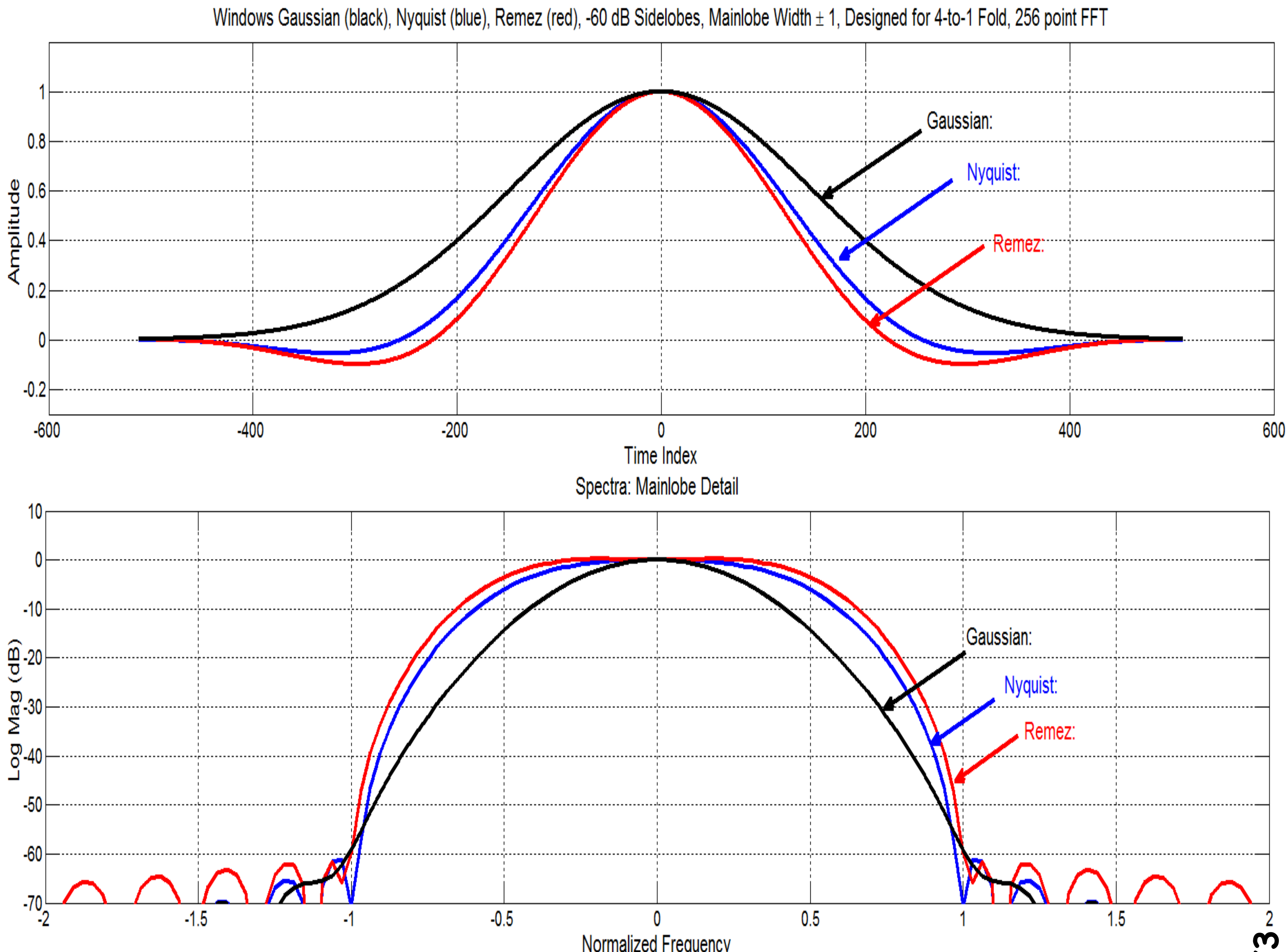


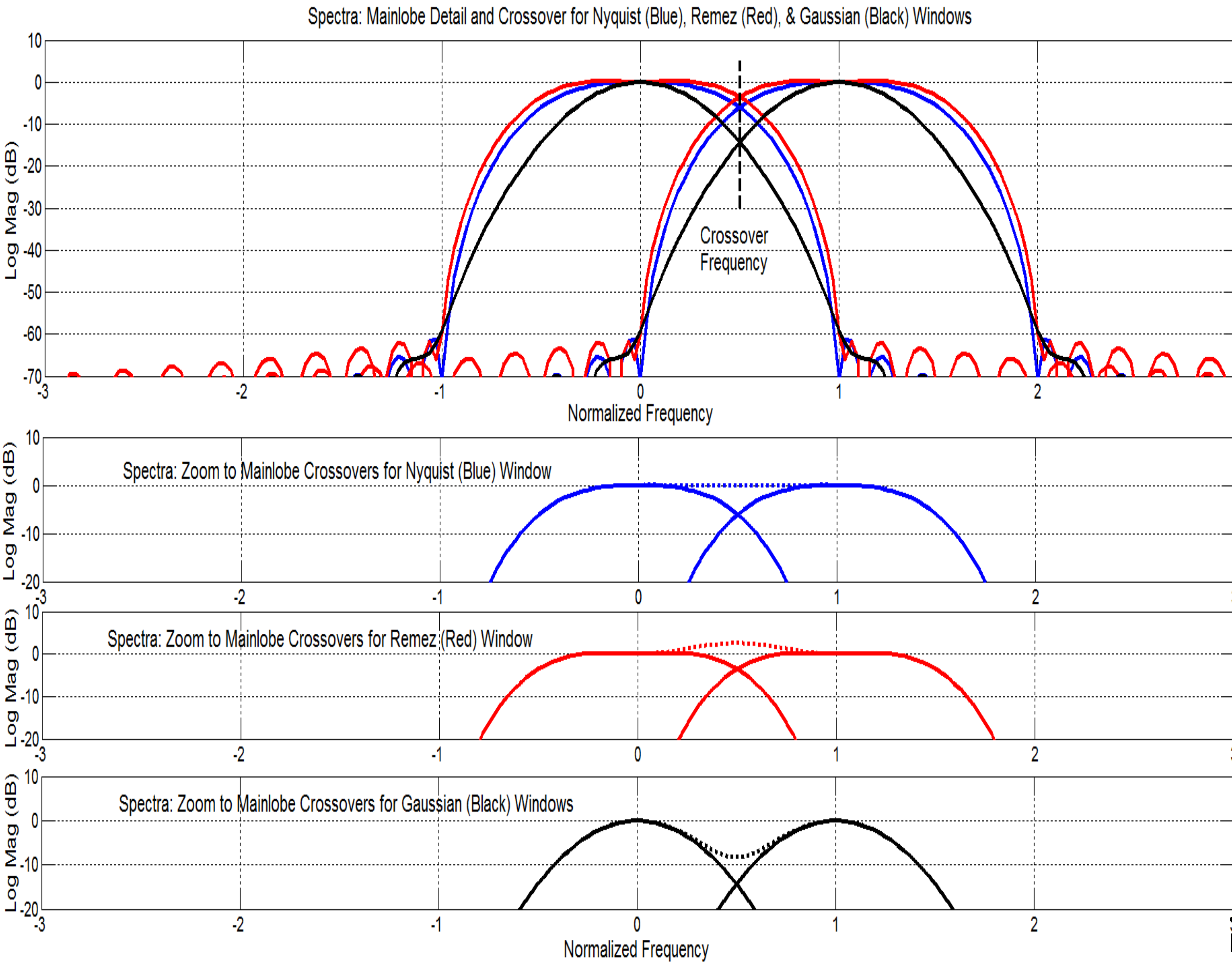
OL	0/8	1/8	2/8	3/8	4/8	5/8	6/8	7/8
N_{AVG}	32	36	42	51	64	85	128	256

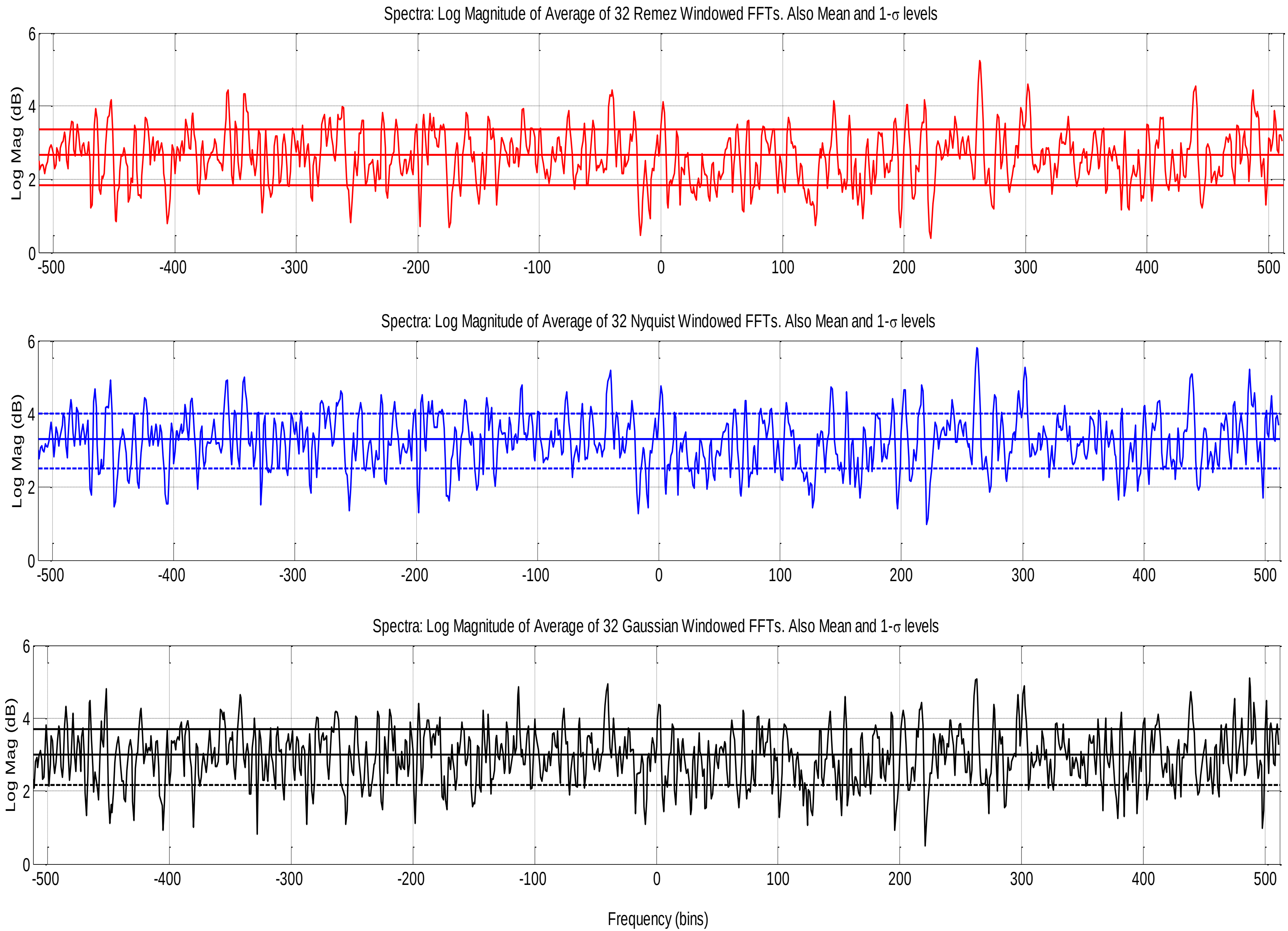


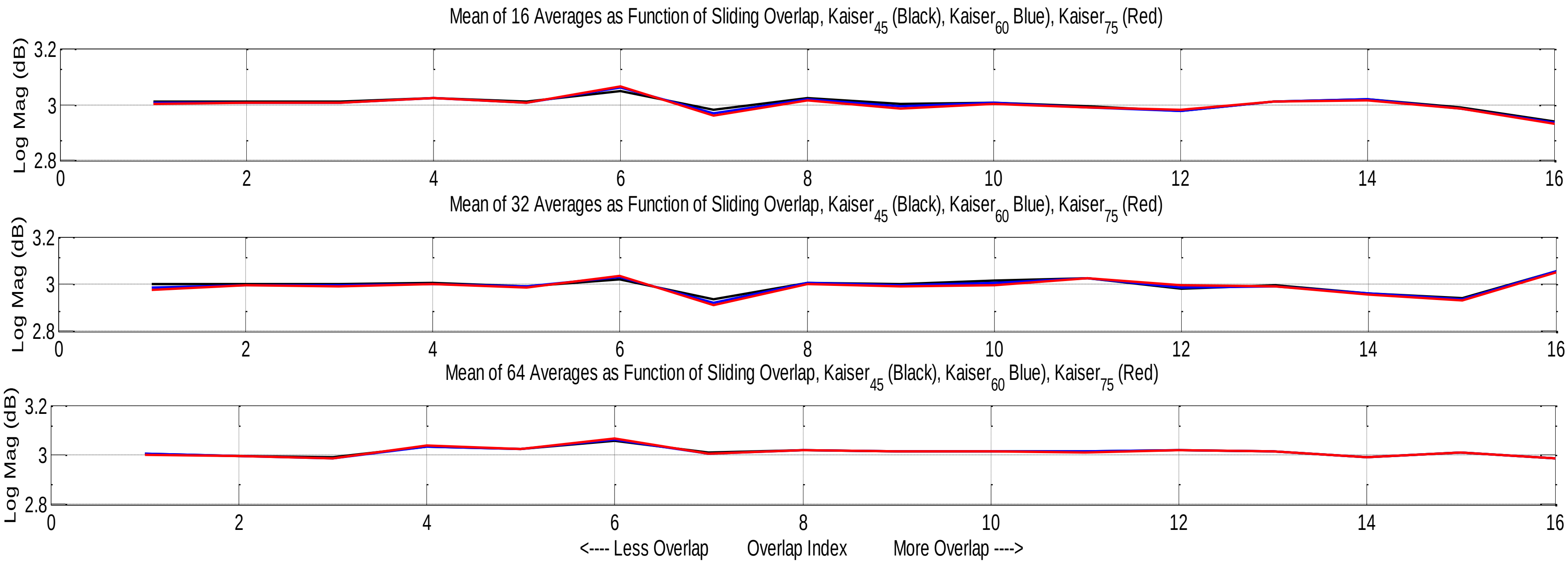
Compare
Effective Number of Independent Measurements
to ENBW of Various Windows.

WINDOW	N_eff(Sat)	ENBW	RATIO
Gaussian	97	2.123	45.7
Blackman-harris	91	2.004	45.4
Kaiser (-80 dB)	86	1.903	45.2
Dolph-Tchebyshev (-80 dB)	78	1.743	44.7
Hann	67	1.500	44.7
Hamming	61	1.364	44.7
Triangle	60	1.333	45.0

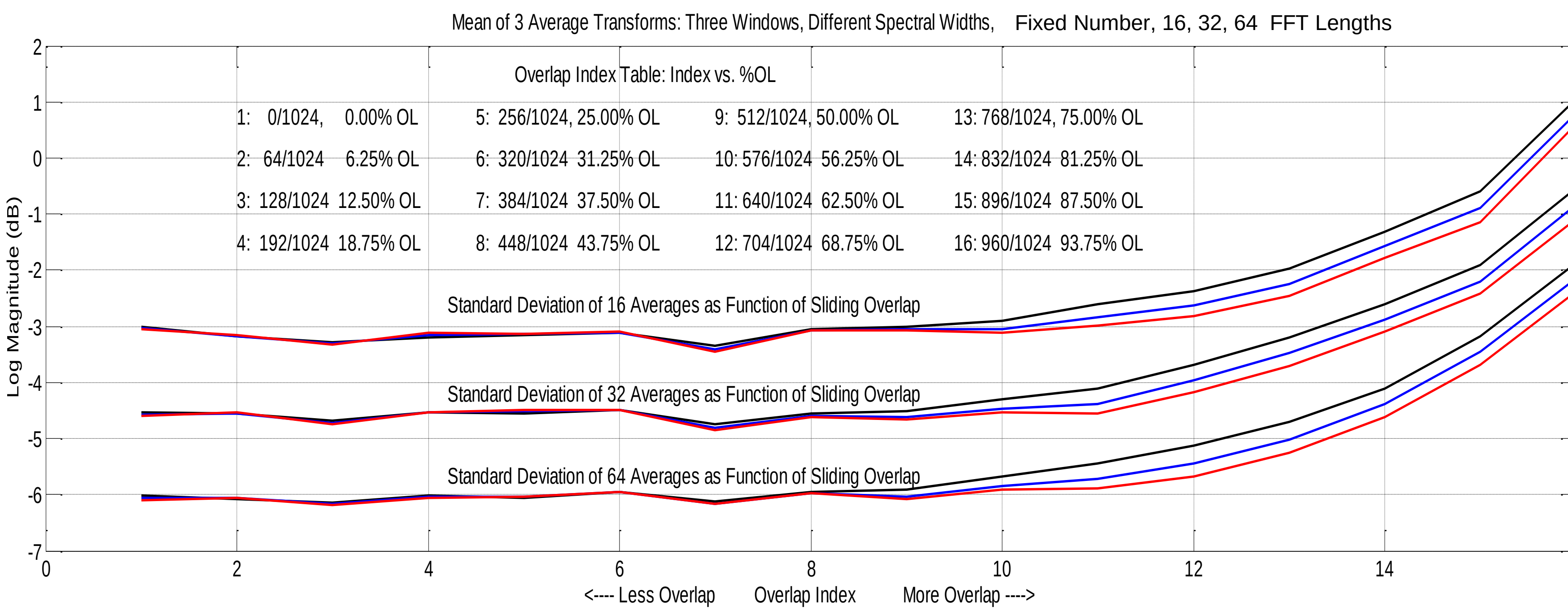

Windows Gaussian (black), Nyquist (blue), Remez (red), -60 dB Sidelobes, Mainlobe Width ± 1, Designed for 4-to-1 Fold, 256 point FFT
Amplitude
Time Index
Gaussian:
Nyquist:
Remez:
Spectra: Mainlobe Detail
Log Mag (dB)
Normalized Frequency
Gaussian:
Nyquist:
Remez:


Spectra: Mainlobe Detail and Crossover for Nyquist (Blue), Remez (Red), & Gaussian (Black) Windows
Log Mag (dB)
Crossover Frequency
Normalized Frequency
Spectra: Zoom to Mainlobe Crossovers for Nyquist (Blue) Window
Log Mag (dB)
Spectra: Zoom to Mainlobe Crossovers for Remez (Red) Window
Log Mag (dB)
Spectra: Zoom to Mainlobe Crossovers for Gaussian (Black) Windows
Log Mag (dB)
Normalized Frequency


Spectra: Log Magnitude of Average of 32 Remez Windowed FFTs. Also Mean and 1-σ levels
Log Mag (dB)
Spectra: Log Magnitude of Average of 32 Nyquist Windowed FFTs. Also Mean and 1-σ levels
Log Mag (dB)
Spectra: Log Magnitude of Average of 32 Gaussian Windowed FFTs. Also Mean and 1-σ levels
Log Mag (dB)
Frequency (bins)

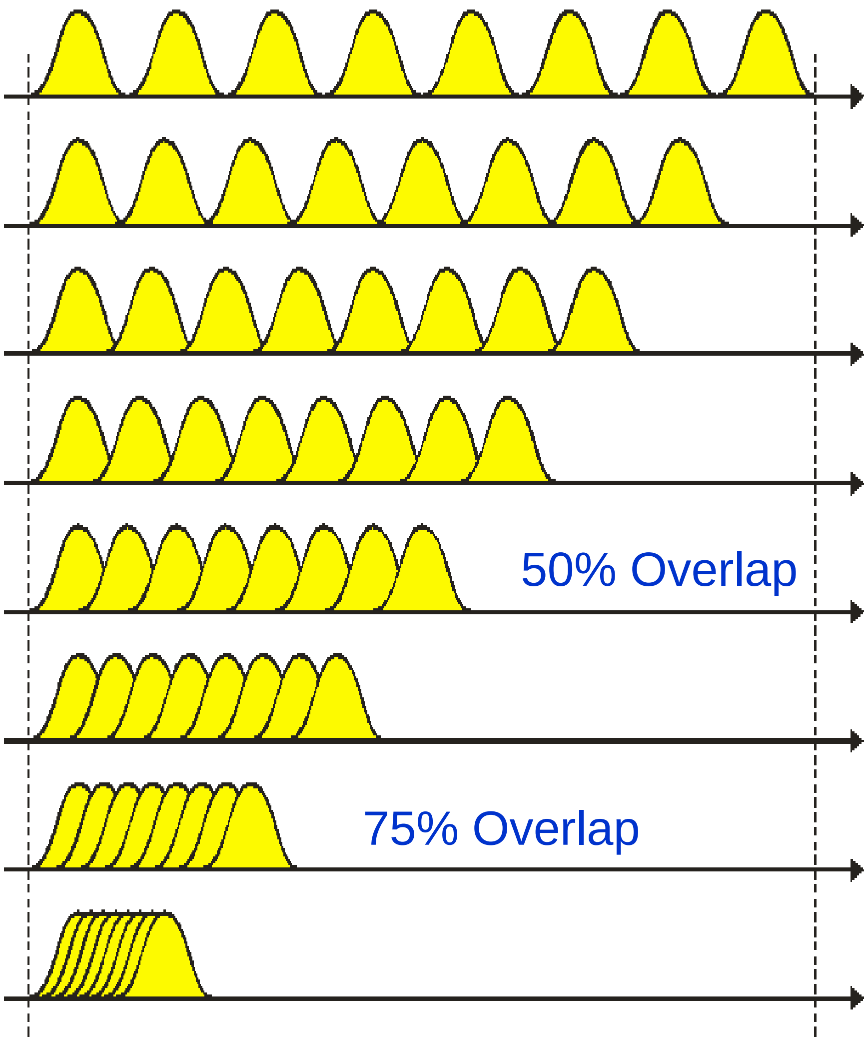


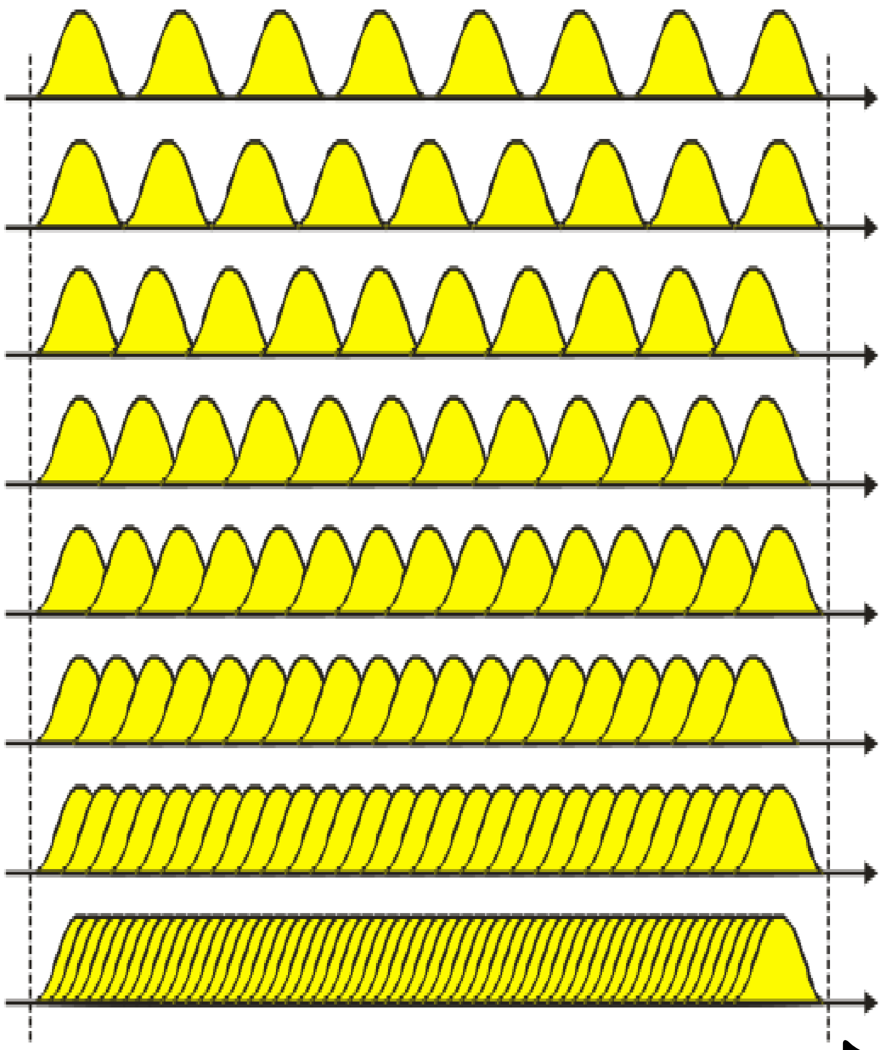
Mean of 16 Averages as Function of Sliding Overlap, Kaiser45 (Black), Kaiser60 Blue), Kaiser75 (Red)
Mean of 32 Averages as Function of Sliding Overlap, Kaiser45 (Black), Kaiser60 Blue), Kaiser75 (Red)
Mean of 64 Averages as Function of Sliding Overlap, Kaiser45 (Black), Kaiser60 Blue), Kaiser75 (Red)
Log Mag (dB)
<---- Less Overlap
Overlap Index
More Overlap ---->

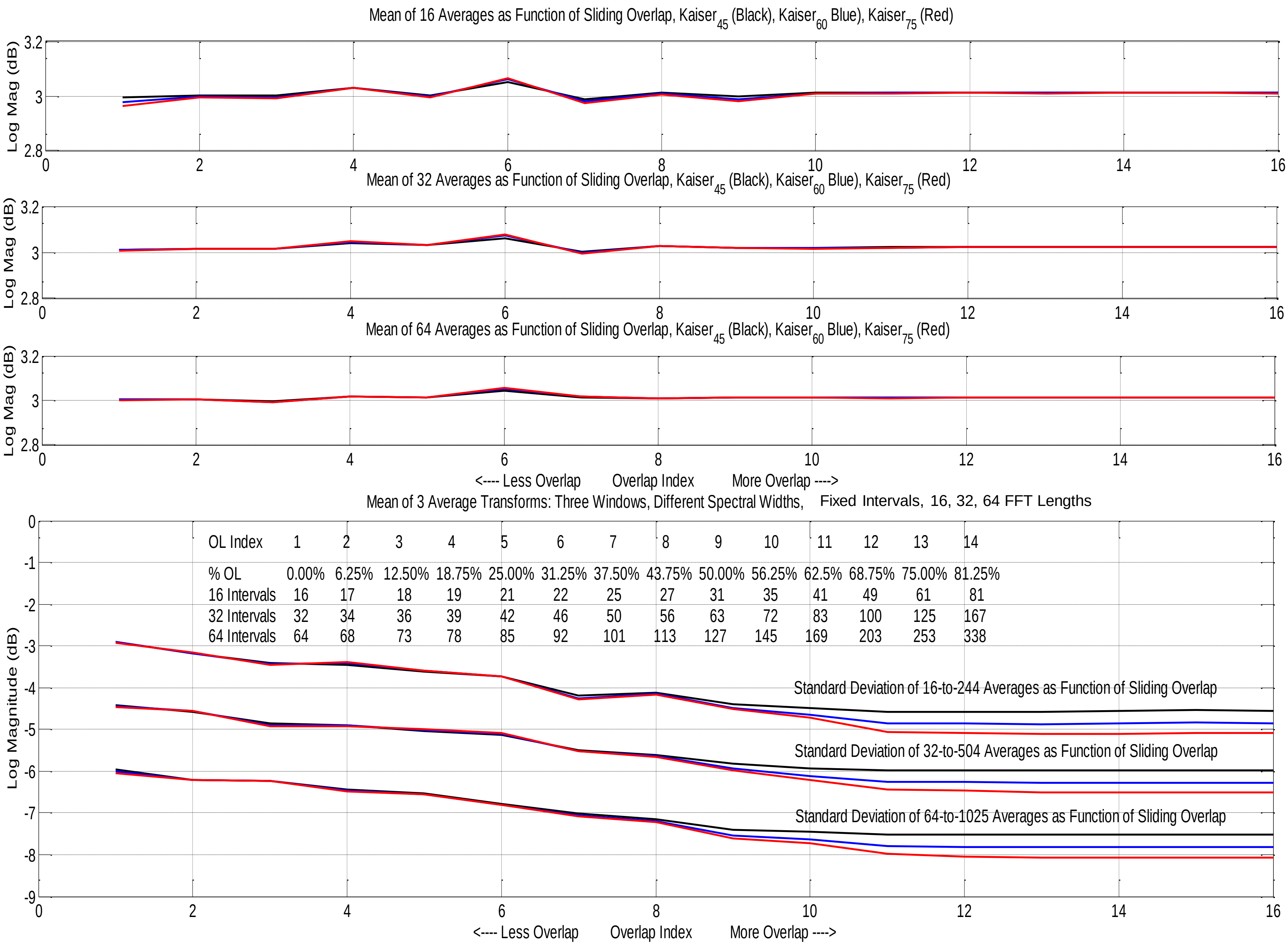


Mean of 3 Average Transforms: Three Windows, Different Spectral Widths, Fixed Number, 16, 32, 64 FFT Lengths
Overlap Index Table: Index vs. %OL
1: 0/1024, 0.00% OL
2: 64/1024 6.25% OL
3: 128/1024 12.50% OL
4: 192/1024 18.75% OL
5: 256/1024, 25.00% OL
6: 320/1024 31.25% OL
7: 384/1024 37.50% OL
8: 448/1024 43.75% OL
9: 512/1024, 50.00% OL
10: 576/1024 56.25% OL
11: 640/1024 62.50% OL
12: 704/1024 68.75% OL
13: 768/1024, 75.00% OL
14: 832/1024 81.25% OL
15: 896/1024 87.50% OL
16: 960/1024 93.75% OL
Standard Deviation of 16 Averages as Function of Sliding Overlap
Standard Deviation of 32 Averages as Function of Sliding Overlap
Standard Deviation of 64 Averages as Function of Sliding Overlap
Log Magnitude (dB)
<---- Less Overlap
Overlap Index
More Overlap ---->

FIXED NUMBER OF OVERLAPPED WINDOWS, ACQUIRE ADDITIONAL DEGREES OF FREEDOM IN SHORTER TIME SPAN
FIXED INTERVAL, INCREASED NUMBER OF OVERLAPPED WINDOWS ACQUIRE ADDITIONAL DEGREES OF FREEDOM






Mean of 16 Averages as Function of Sliding Overlap, Kaiser45 (Black), Kaiser60 Blue), Kaiser75 (Red)
Mean of 32 Averages as Function of Sliding Overlap, Kaiser45 (Black), Kaiser60 Blue), Kaiser75 (Red)
Mean of 64 Averages as Function of Sliding Overlap, Kaiser45 (Black), Kaiser60 Blue), Kaiser75 (Red)
Log Mag (dB)
<---- Less Overlap Overlap Index More Overlap ---->
Mean of 3 Average Transforms: Three Windows, Different Spectral Widths, Fixed Intervals, 16, 32, 64 FFT Lengths
OL Index 1 2 3 4 5 6 7 8 9 10 11 12 13 14
% OL 0.00% 6.25% 12.50% 18.75% 25.00% 31.25% 37.50% 43.75% 50.00% 56.25% 62.5% 68.75% 75.00% 81.25%
16 Intervals 16 17 18 19 21 22 25 27 31 35 41 49 61 81
32 Intervals 32 34 36 39 42 46 50 56 63 72 83 100 125 167
64 Intervals 64 68 73 78 85 92 101 113 127 145 169 203 253 338
Standard Deviation of 16-to-244 Averages as Function of Sliding Overlap
Standard Deviation of 32-to-504 Averages as Function of Sliding Overlap
Standard Deviation of 64-to-1025 Averages as Function of Sliding Overlap
Log Magnitude (dB)
<---- Less Overlap Overlap Index More Overlap ---->

Resolution Versus Uncertainty

Optimum Integration Time

Resolution = Uncertainty

$$res = \frac{1}{\mathrm{T}} = ST = \text{uncertainty},$$

$$\frac{1}{T^2} = S,$$

$$T^2 = \frac{1}{S},$$

$$T = \frac{1}{\sqrt{S}}$$

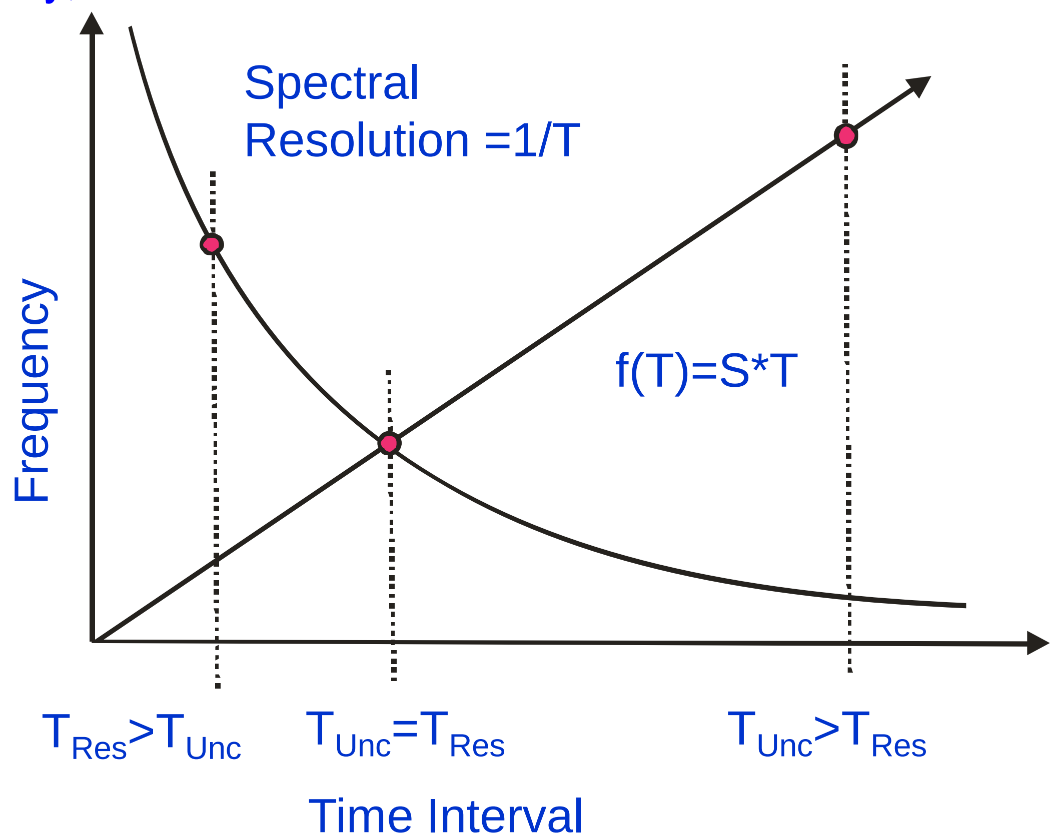
Ex. if S = 10 Hz/sec,

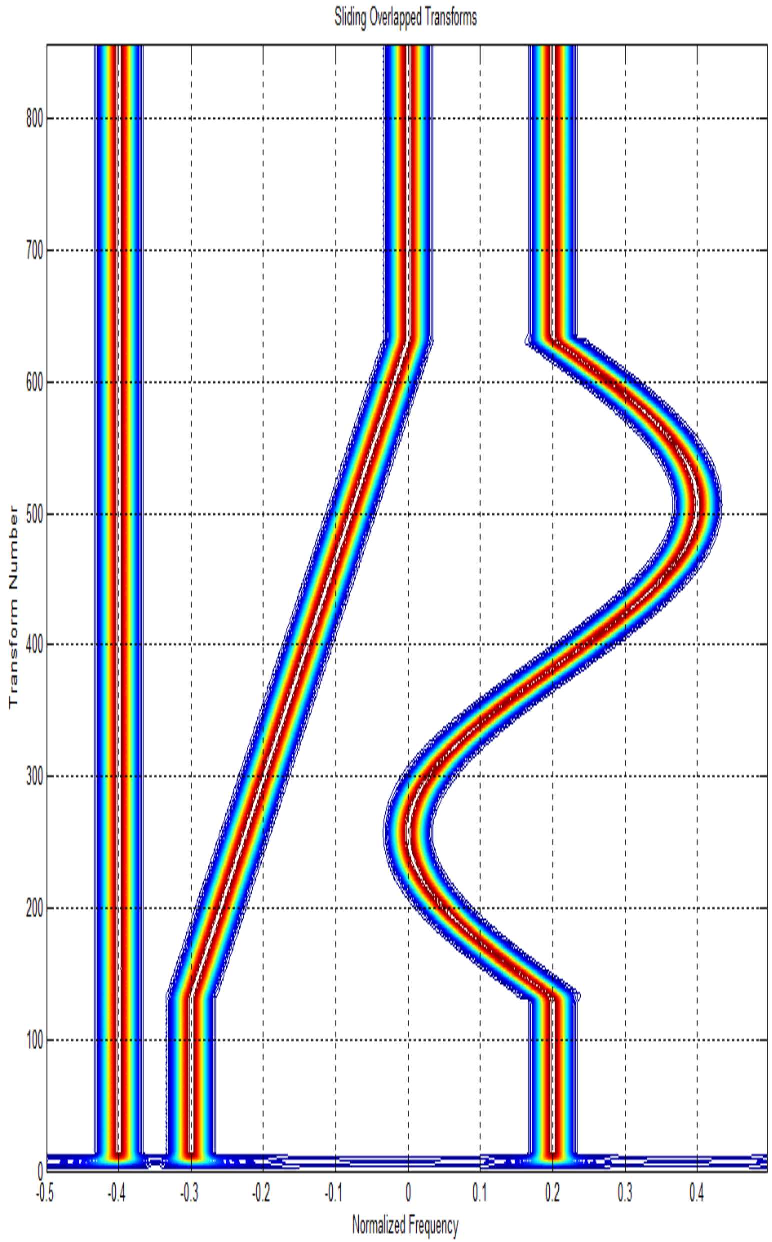
$$\mathrm{T} = \frac{1}{3.16} = 0.316$$

Res = 1/T = 3.16 Hz

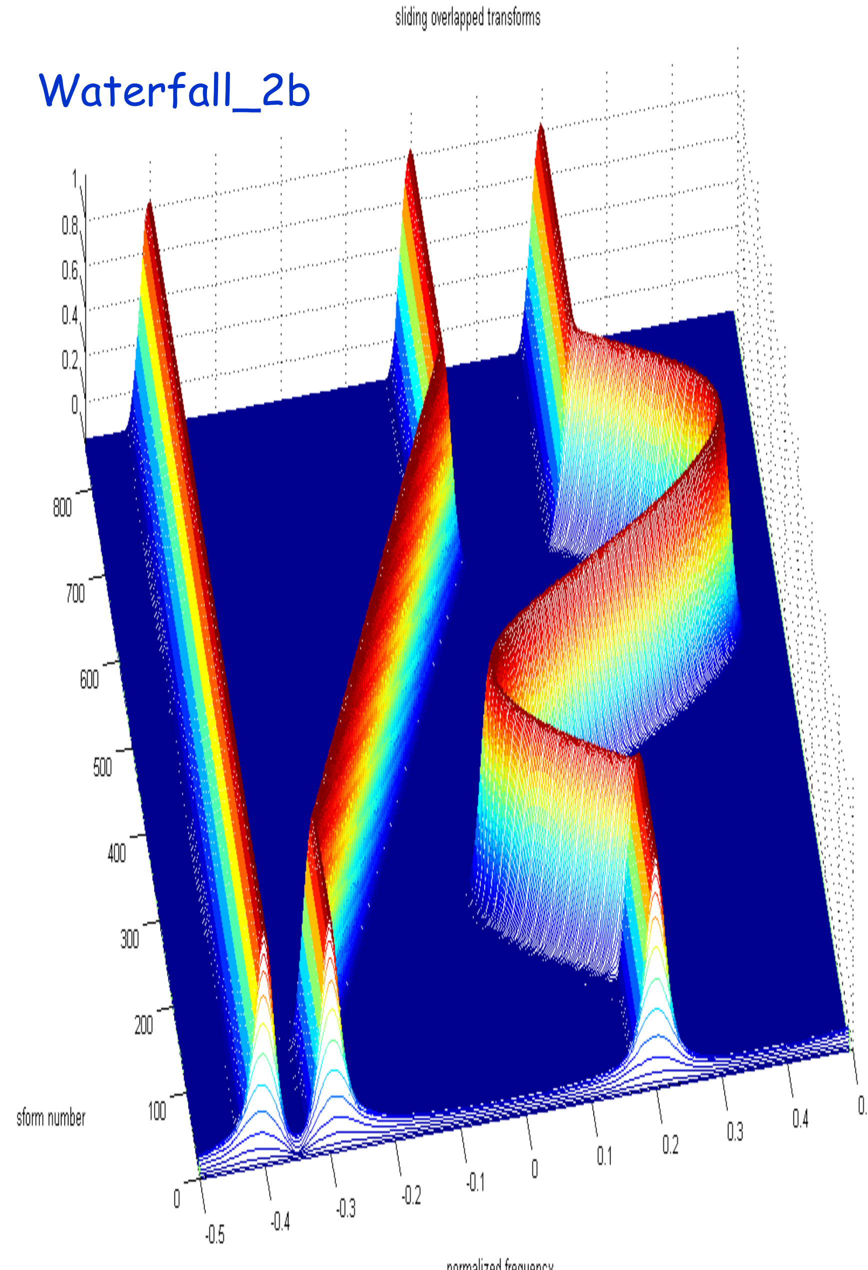
FM Sweep
Spectral Uncertainty =ST.
Units of S: Rad/sec/sec

Non
Stationary
Signals

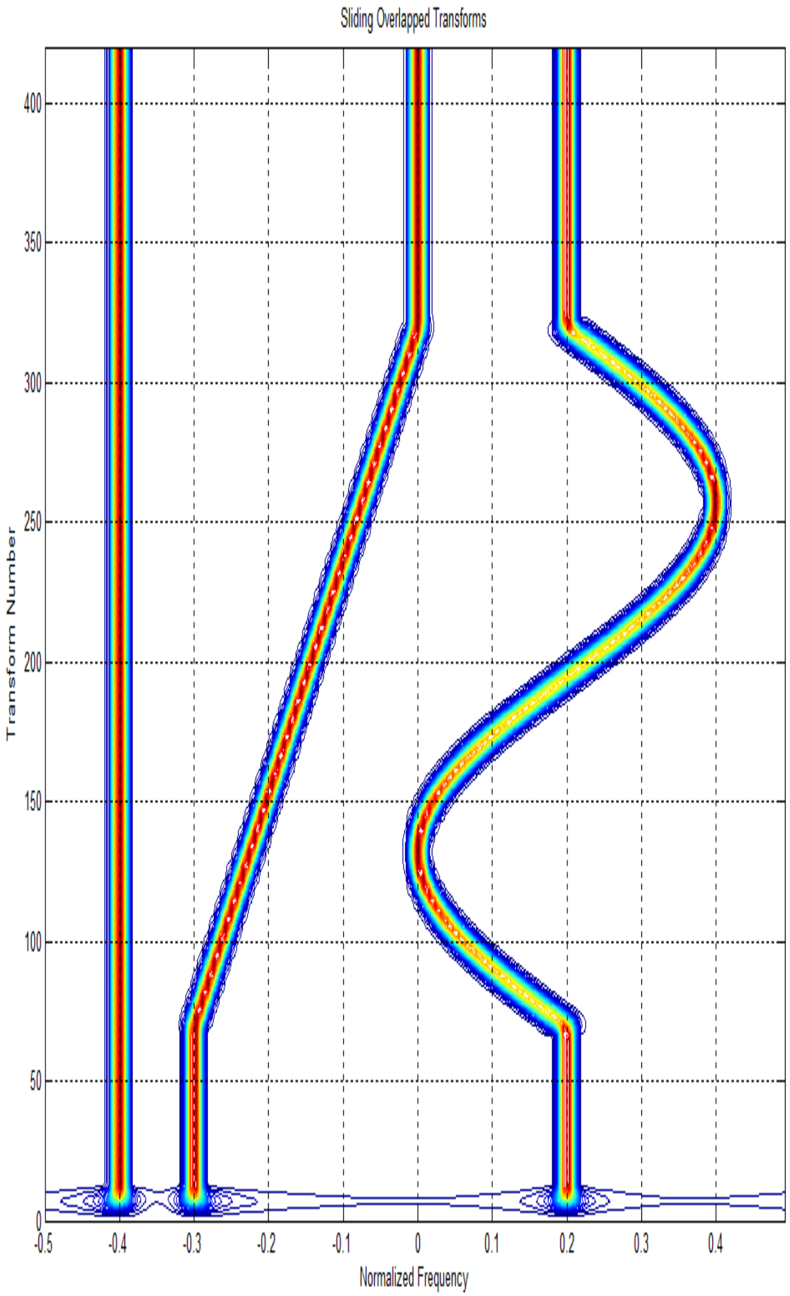


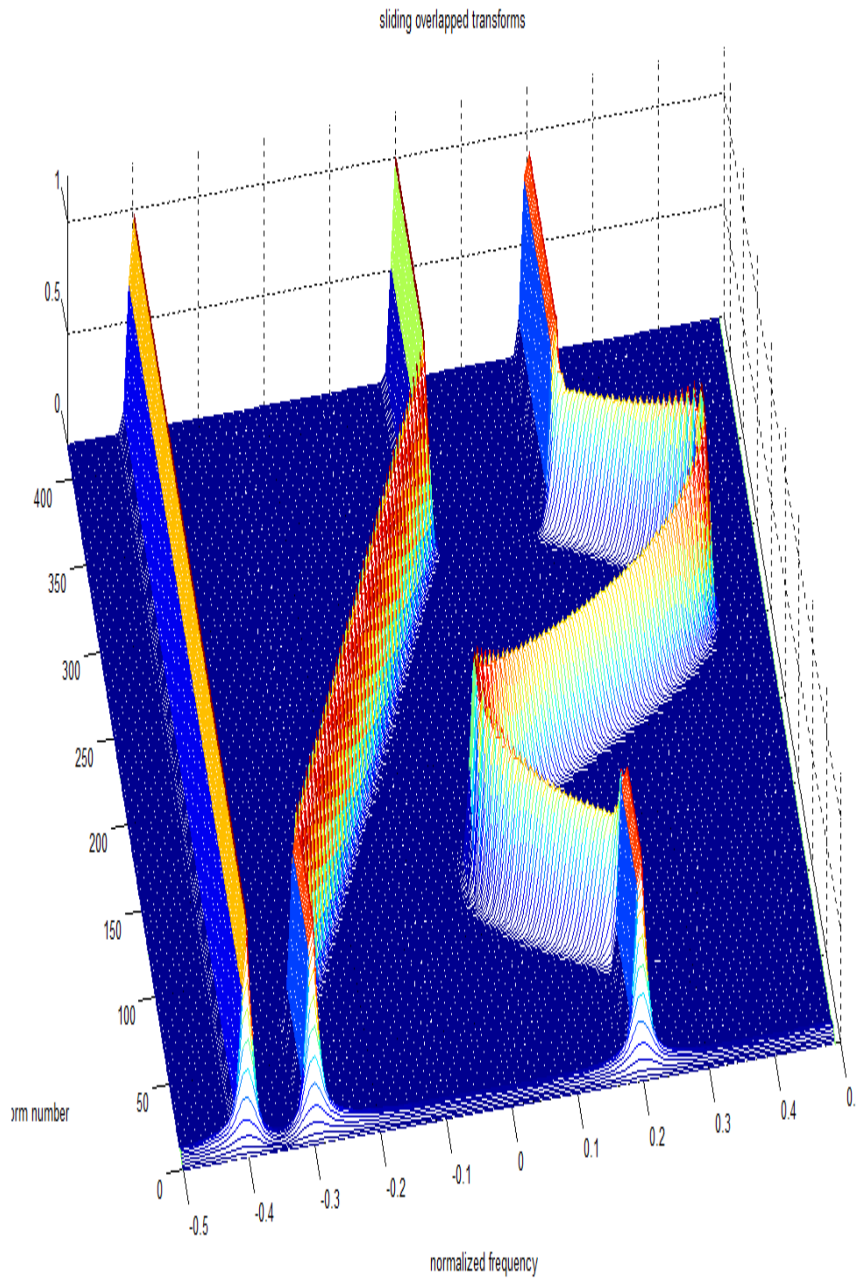


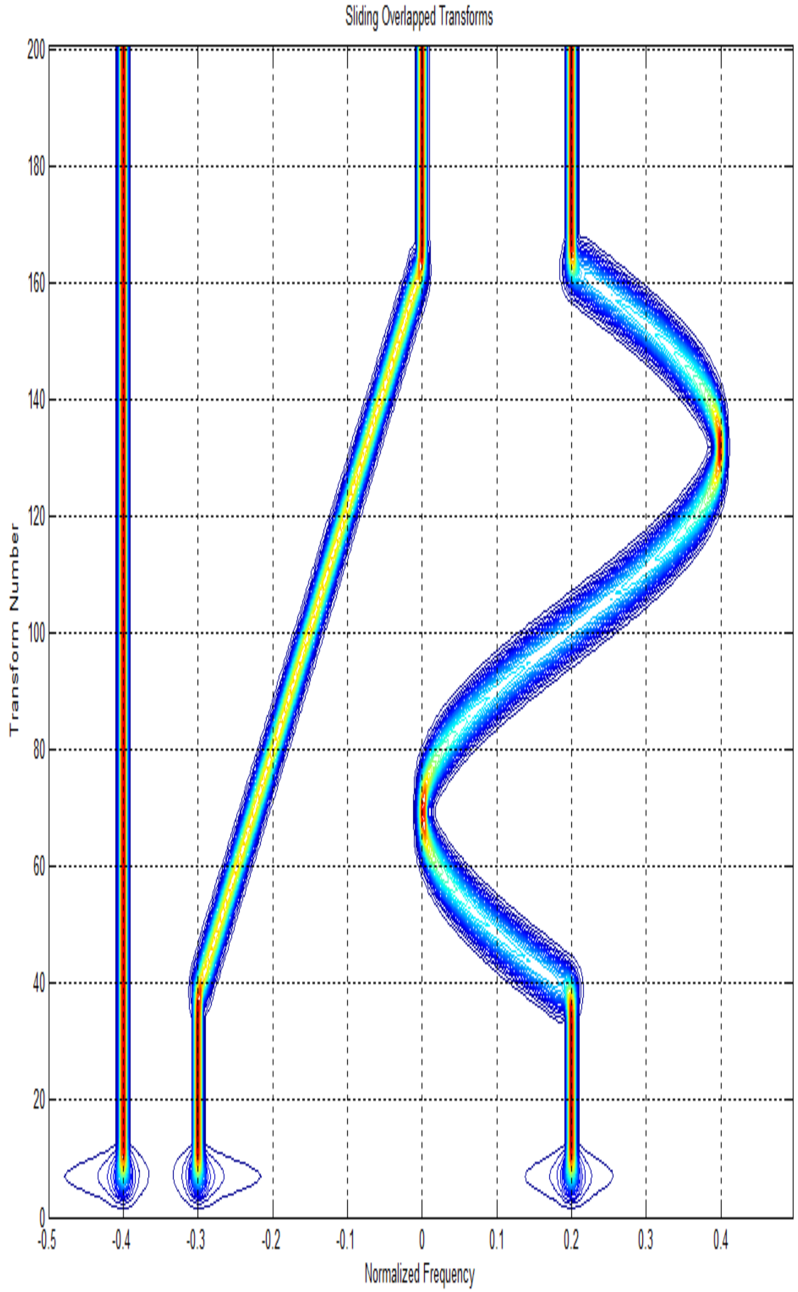
Sliding Overlapped Transforms
Transform Number
Normalized Frequency
0
100
200
300
400
500
600
700
800
-0.5
-0.4
-0.3
-0.2
-0.1
0
0.1
0.2
0.3
0.4

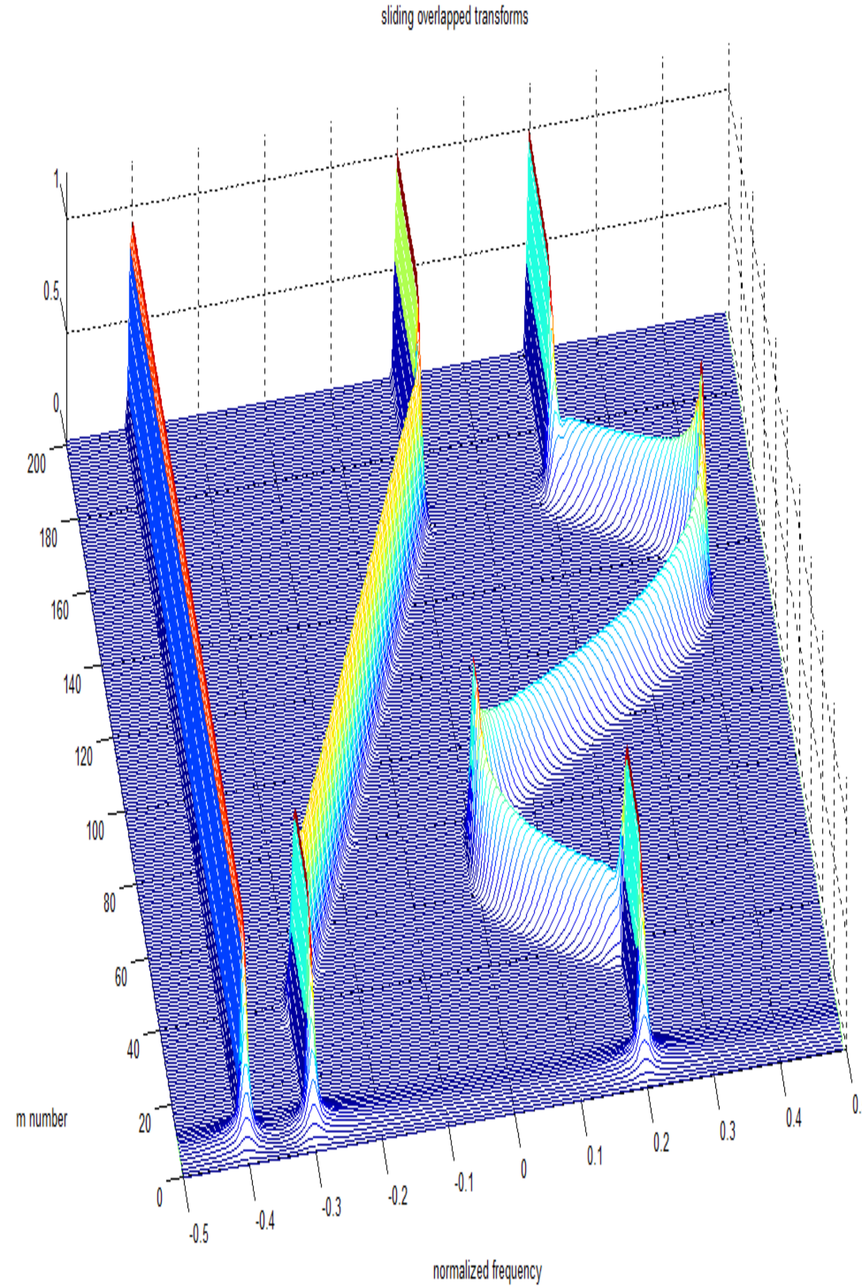


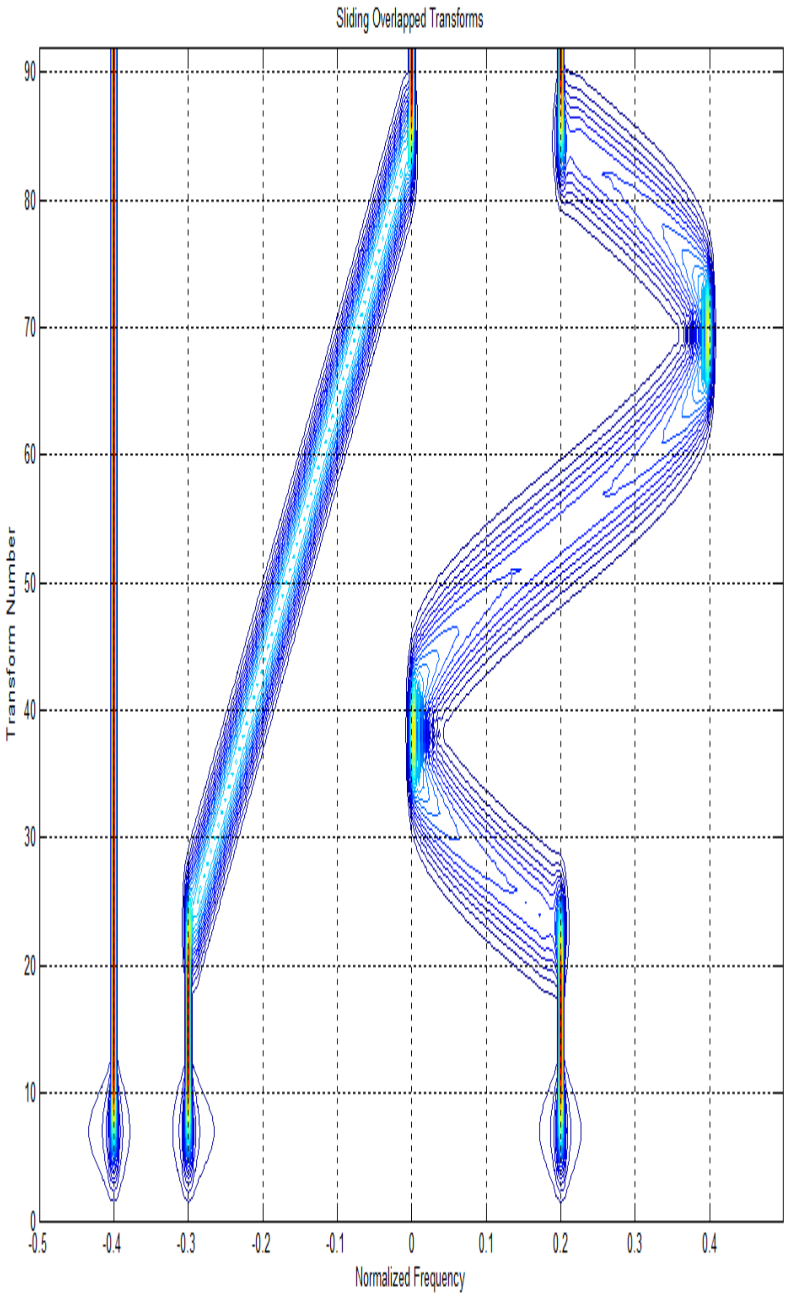
sliding overlapped transforms
Waterfall_2b
sform number
normalized frequency


Sliding Overlapped Transforms
Transform Number
Normalized Frequency
0 50 100 150 200 250 300 350 400
-0.5 -0.4 -0.3 -0.2 -0.1 0 0.1 0.2 0.3 0.4

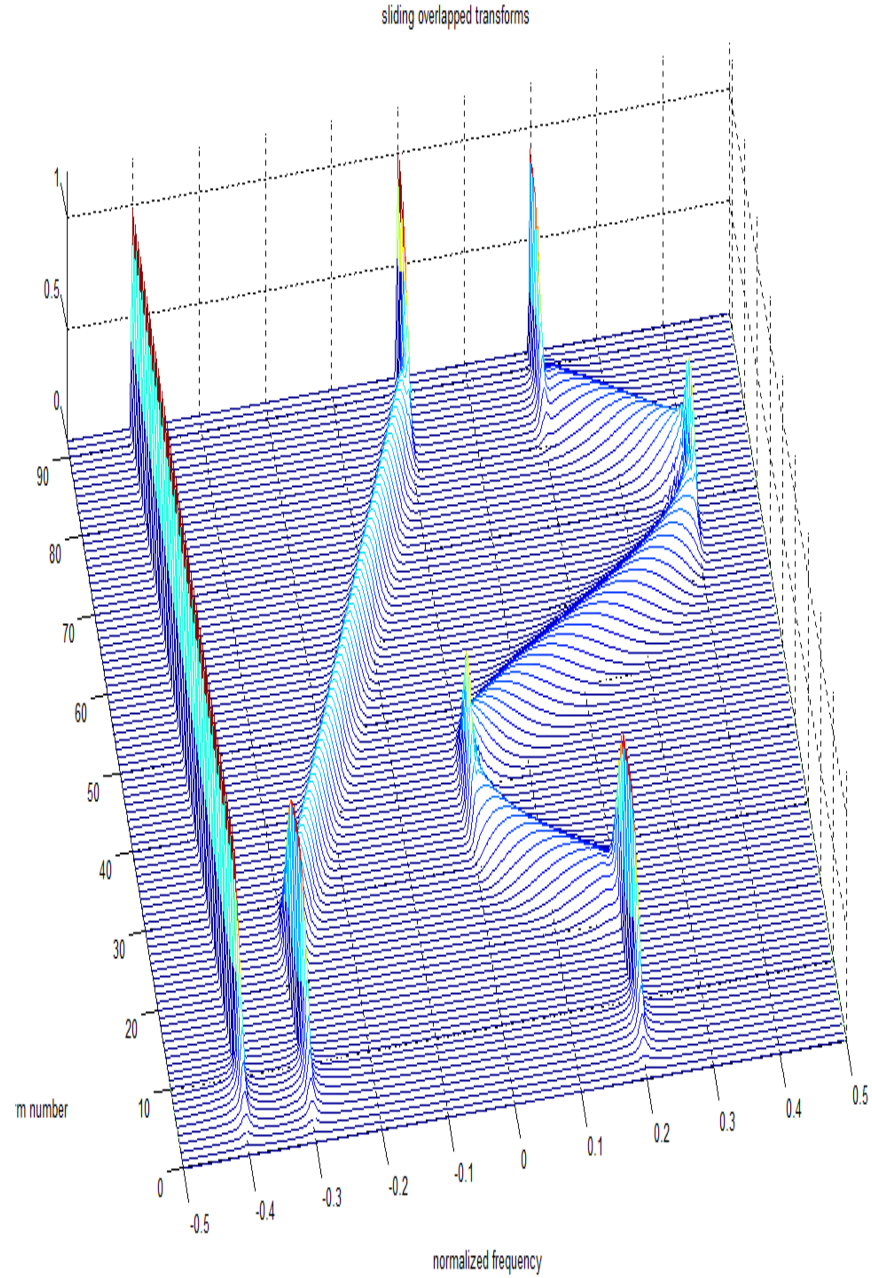

sliding overlapped transforms
normalized frequency
1
0.5
0
0 50 100 150 200 250 300 350 400
-0.5 -0.4 -0.3 -0.2 -0.1 0 0.1 0.2 0.3 0.4 0.5


Sliding Overlapped Transforms
Transform Number
Normalized Frequency
0
20
40
60
80
100
120
140
160
180
200
-0.5
-0.4
-0.3
-0.2
-0.1
0
0.1
0.2
0.3
0.4

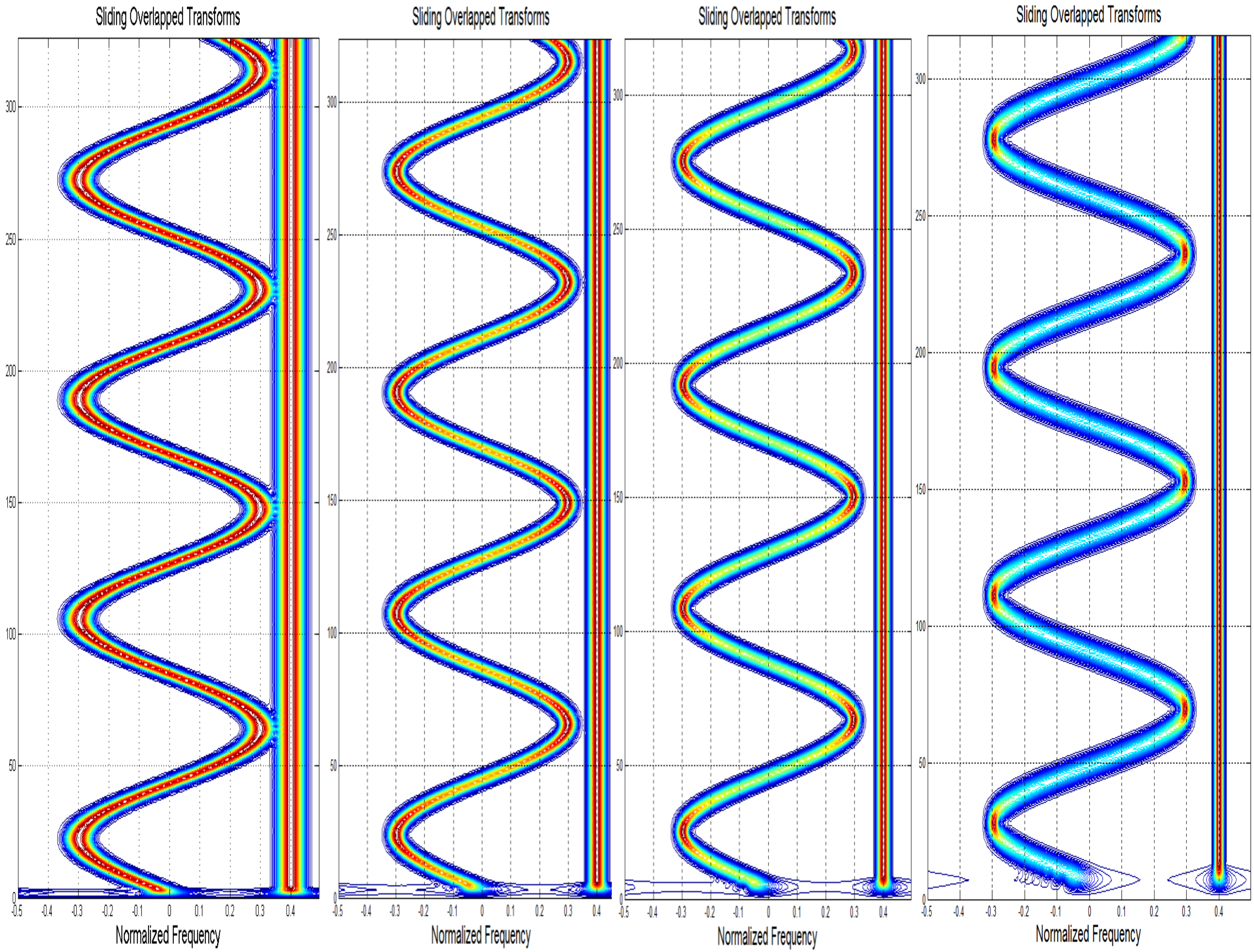

sliding overlapped transforms
1
0.5
0
m number
normalized frequency
0
20
40
60
80
100
120
140
160
180
200
-0.5
-0.4
-0.3
-0.2
-0.1
0
0.1
0.2
0.3
0.4
0.5



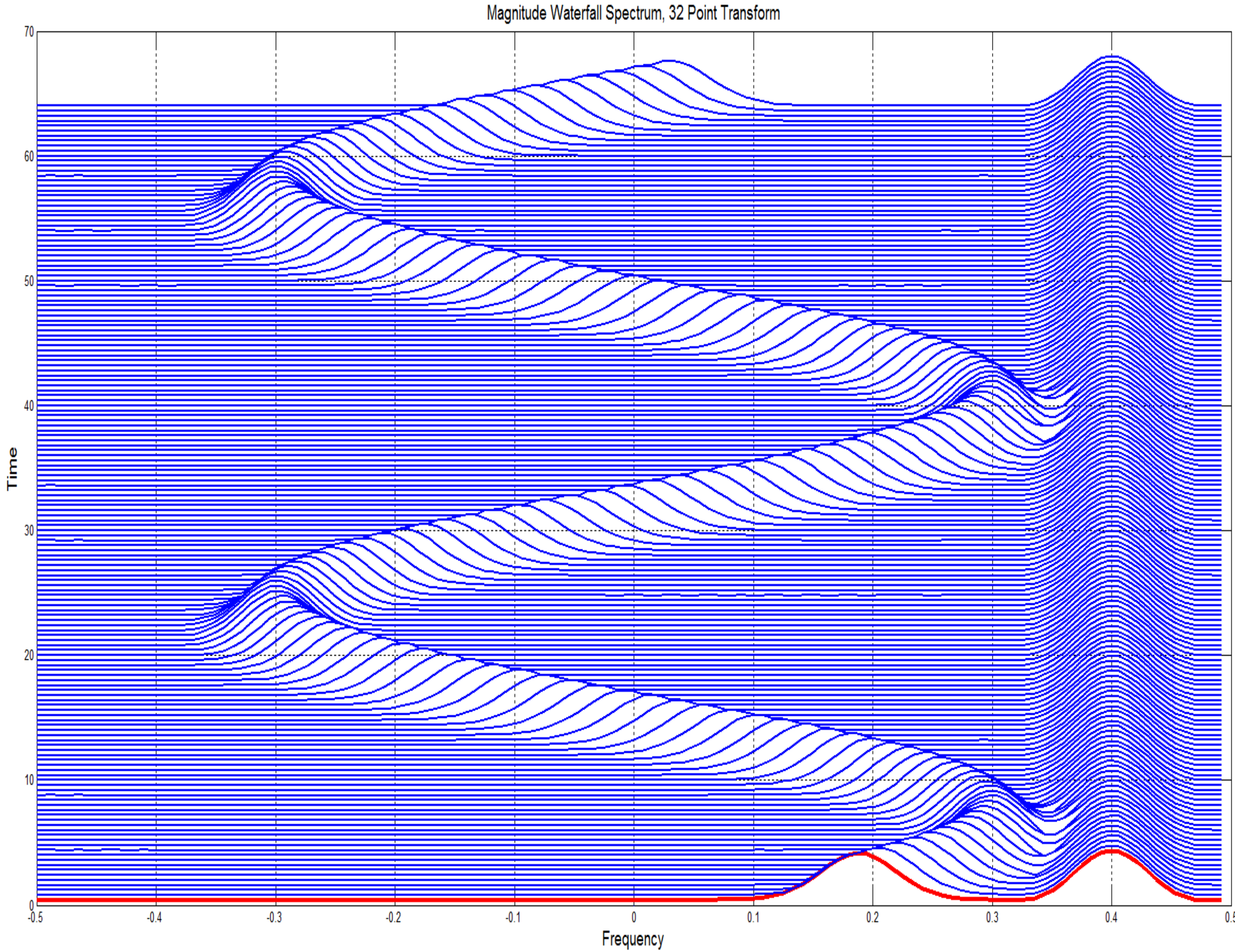
Sliding Overlapped Transforms
Transform Number
Normalized Frequency



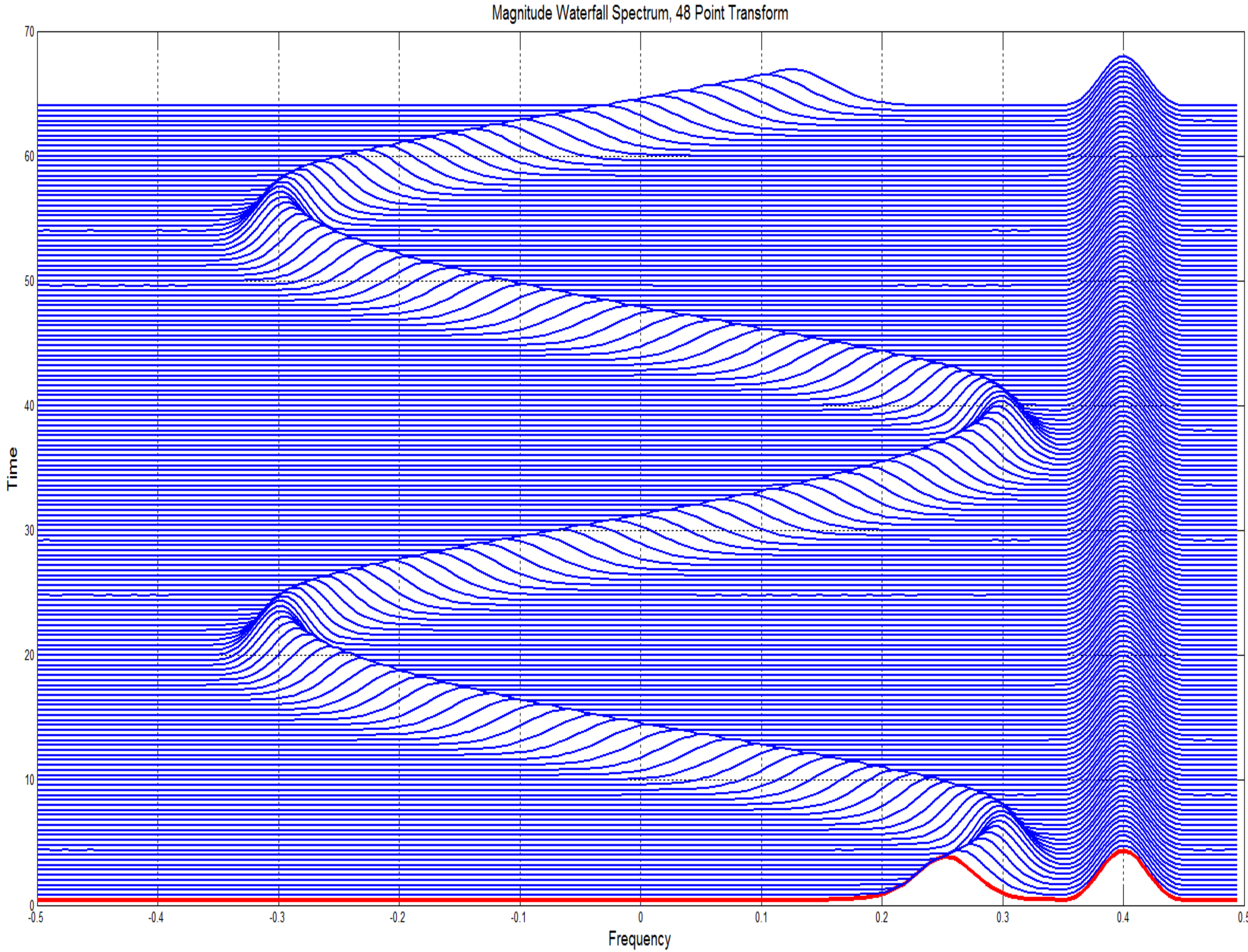
sliding overlapped transforms
normalized frequency

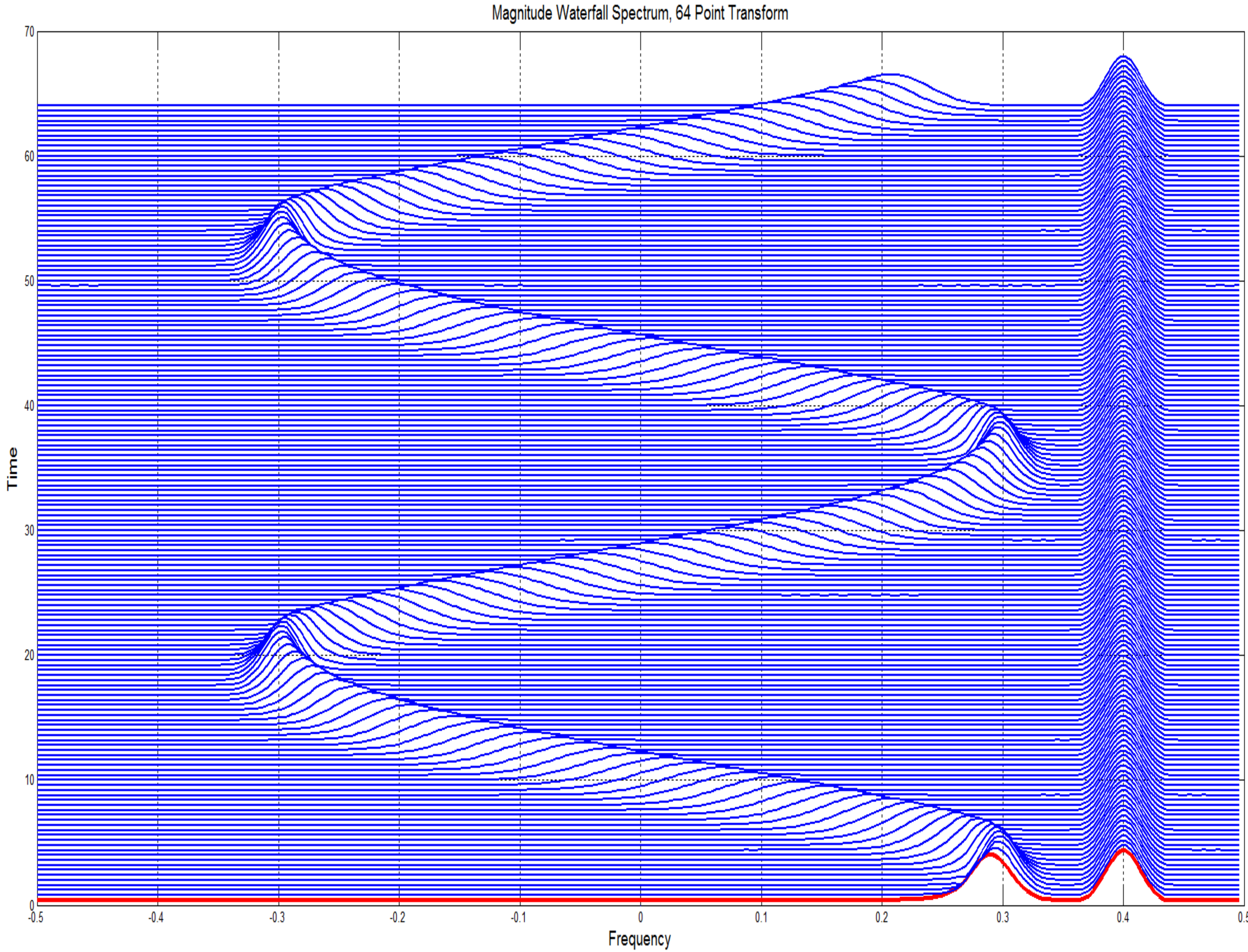


Sliding Overlapped Transforms
Sliding Overlapped Transforms
Sliding Overlapped Transforms
Sliding Overlapped Transforms
Normalized Frequency
Normalized Frequency
Normalized Frequency
Normalized Frequency

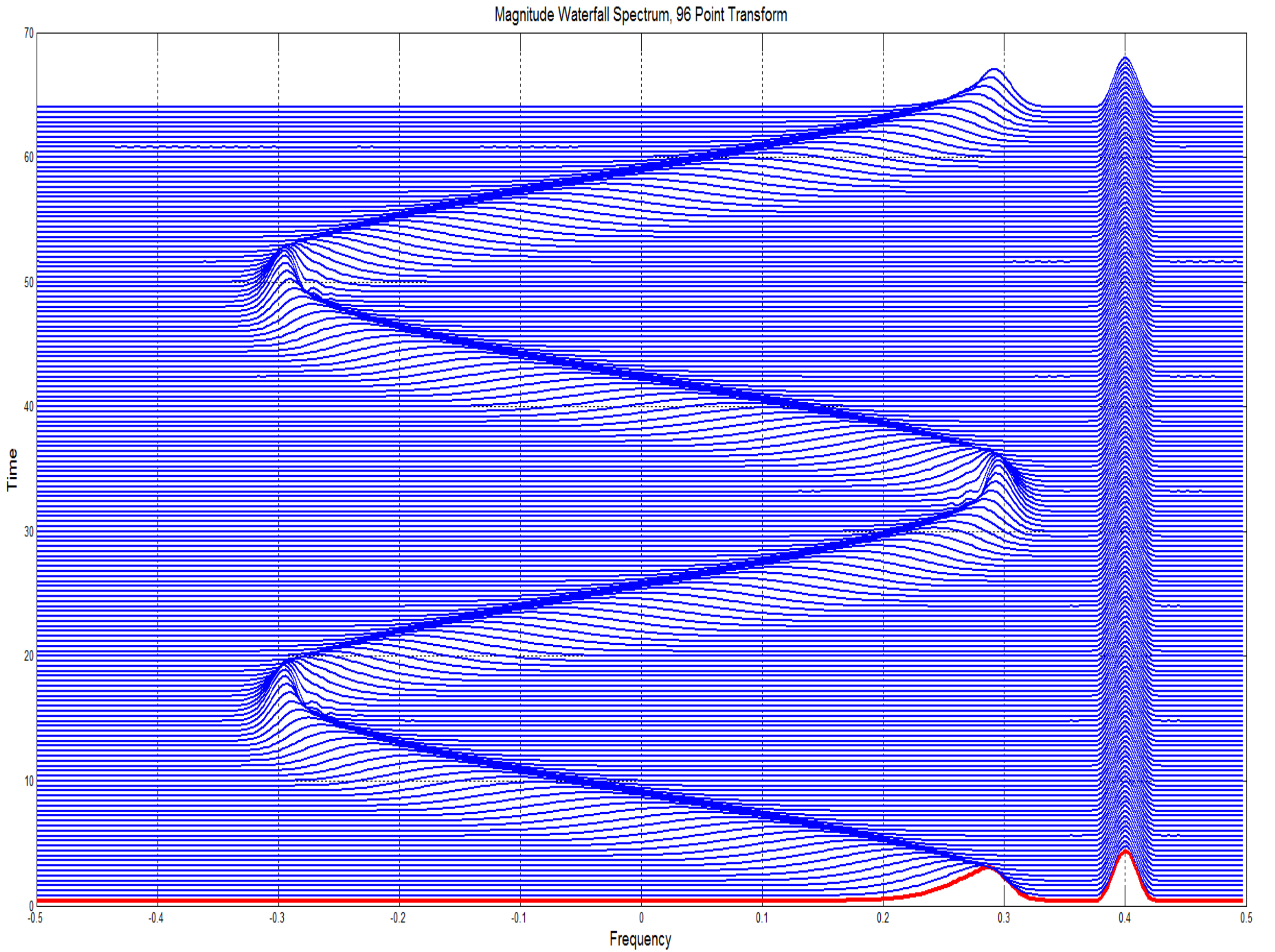


Magnitude Waterfall Spectrum, 32 Point Transform
Time
Frequency
70
60
50
40
30
20
10
0
-0.5
-0.4
-0.3
-0.2
-0.1
0
0.1
0.2
0.3
0.4
0.5


Magnitude Waterfall Spectrum, 48 Point Transform
Time
Frequency
70
60
50
40
30
20
10
0
-0.5
-0.4
-0.3
-0.2
-0.1
0
0.1
0.2
0.3
0.4
0.5



Magnitude Waterfall Spectrum, 64 Point Transform
Time
Frequency
70
60
50
40
30
20
10
0
-0.5
-0.4
-0.3
-0.2
-0.1
0
0.1
0.2
0.3
0.4
0.5



Magnitude Waterfall Spectrum, 96 Point Transform
Time
Frequency
70
60
50
40
30
20
10
0
-0.5
-0.4
-0.3
-0.2
-0.1
0
0.1
0.2
0.3
0.4
0.5

FINALLY!
THE DISCRETE FOURIER TRANSFORM
AND
THE FAST FOURIER TRANSFORM

Gauss and the History of the Fast Fourier Transform, IEEE ASSP Magazine, October 1984

Principal Discoveries of Efficient Methods of Computing the DFT

Researcher(s)	Date	Sequence Lengths	Number of DFT Values	Application
C.F. Gauss	1805	Any Composite Integer	All	Interpolation of orbits of celestial bodies
F. Carlinia	1828	12	---	Harmonic analysis of barometric pressure
A. Smith	1846	4, 8, 16, 32	5 or 9	Correcting deviations in compasses on ships
J.D. Everett	1860	12	5	Modeling underground temperature deviations
C. Runge	1903	$2^n k$	All	Harmonic analysis of functions
K. Stumpff	1939	$2^n k$, $3^n k$	All	Harmonic analysis of functions
Danielson & Lanczos	1942	$2^n k$	All	Harmonic analysis of functions
L.H. Thomas	1948	Any Integer with relatively prime factors	All	Harmonic analysis of functions
I.J. Good	1958	Any Integer with relatively prime factors	All	Harmonic analysis of functions
Cooley & Tukey	1965	Any composite Integer	All	Harmonic analysis of functions
S. Winograd	1976	Any Integer with relatively prime factors	All	Complexity Theory for Harmonic analysis

Discrete Fourier Transform

$$F(k) = \sum_{n=0}^{N-1} f(n) e^{-j\frac{2\pi}{N}nk}, k = 0,1,2,\ldots,N-1$$

(DFT)

$$f(n) = \frac{1}{N} \sum_{k=0}^{N-1} F(k) e^{+j\frac{2\pi}{N}nk}, n = 0,1,2,\ldots,N-1$$

(IDFT)

Discrete Fourier Transform Matrix

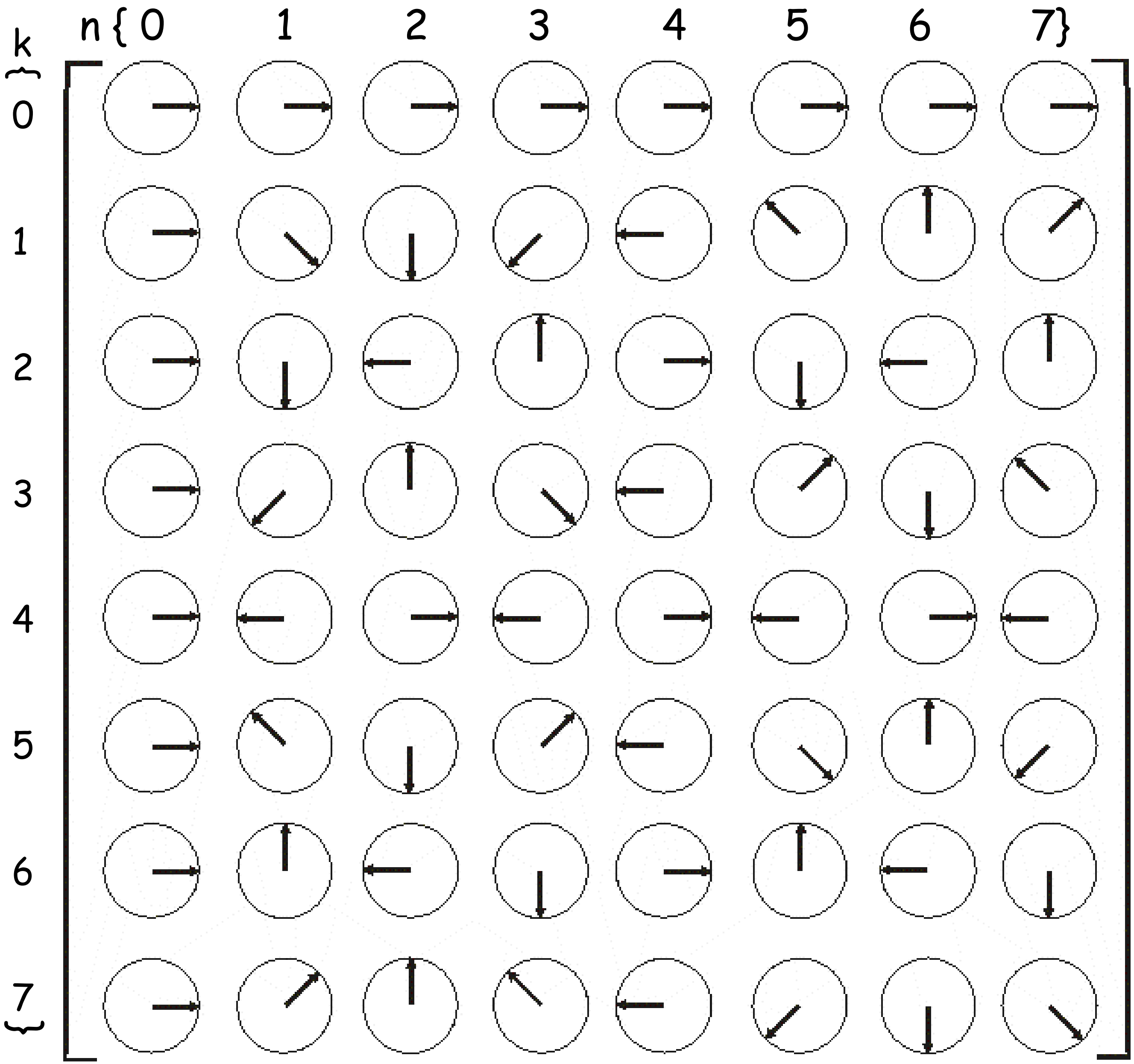
$$
\begin{bmatrix} F_0 \\ F_1 \\ F_2 \\ F_3 \\ \vdots \\ \vdots \\ \vdots \\ F_{N-1} \end{bmatrix}
=
\begin{bmatrix}
w^0 & w^0 & w^0 & w^0 & \cdots & \cdots & \cdots & w^0 \\
w^0 & w^1 & w^2 & w^3 & \cdots & \cdots & \cdots & w^{N-1} \\
w^0 & w^2 & w^4 & w^6 & \cdots & \cdots & \cdots & w^{2(N-1)} \\
w^0 & w^3 & w^6 & w^8 & \cdots & \cdots & \cdots & w^{3(N-1)} \\
\vdots & \vdots & \vdots & \vdots & \ddots & & & \vdots \\
\vdots & \vdots & \vdots & \vdots & & \ddots & & \vdots \\
\vdots & \vdots & \vdots & \vdots & & & \ddots & \vdots \\
w^0 & w^{N-1} & w^{2(N-1)} & w^{3(N-1)} & \cdots & \cdots & \cdots & w^{(N-1)(N-1)}
\end{bmatrix}
\begin{bmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \\ \vdots \\ \vdots \\ \vdots \\ f_{N-1} \end{bmatrix}
$$

$$
\left.\begin{array}{l} \text{N-Operations per Output} \\ \text{N-Outputs per Transform} \end{array}\right\} \; N^2 \text{ Operations per Transform}
$$

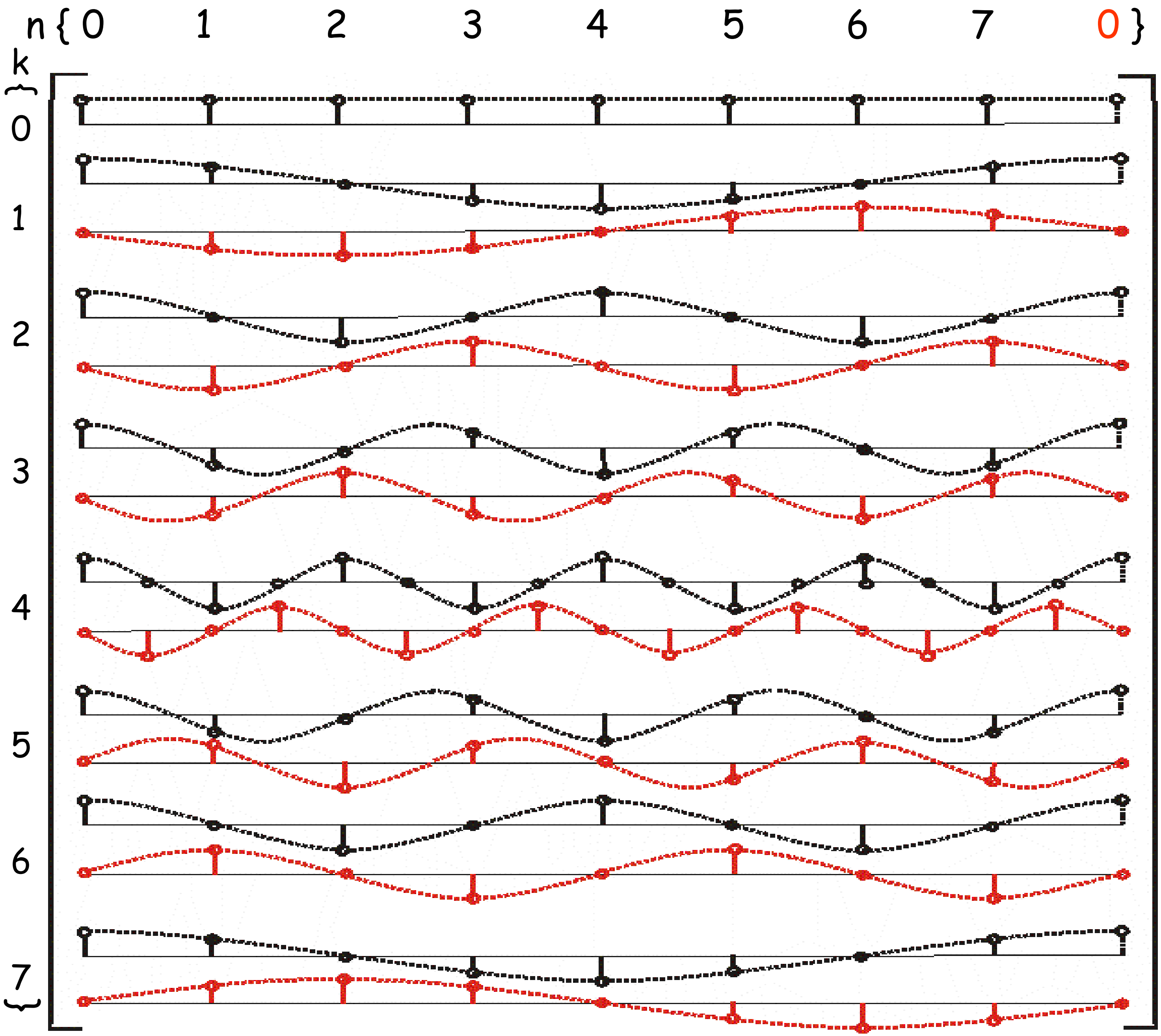
NOTE, This Matrix is its own Transpose

Phasor Representation of DFT Matrix

$$\exp\left(-j\frac{2\pi}{8}nk\right)$$



Cosine + j Sine Representation of DFT Matrix



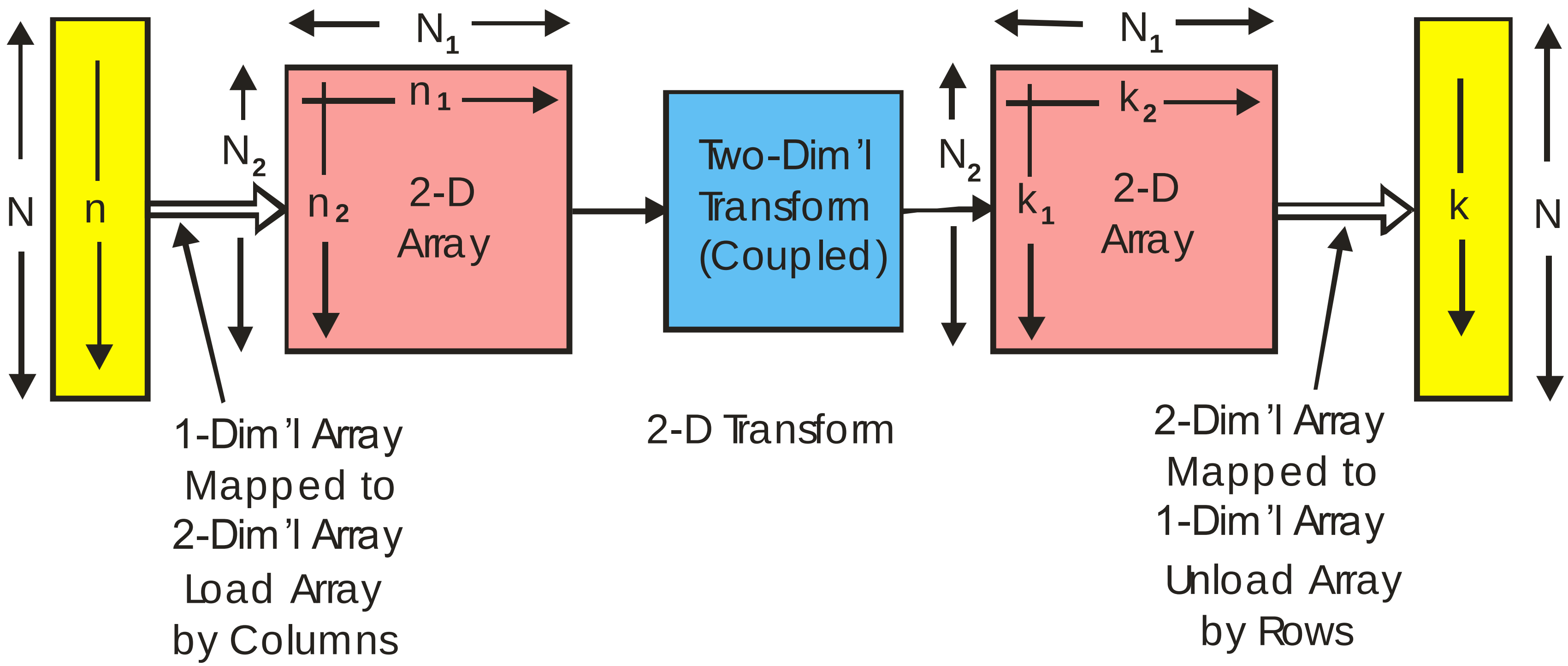
DISCRETE FOURIER TRANSFORM MATRIX IS UNITARY (ALMOST)

$$\overline{F} = W\overline{f} \qquad \text{DFT}$$

$$\overline{f} = W^{-1}\overline{F} \qquad \text{IDFT}$$

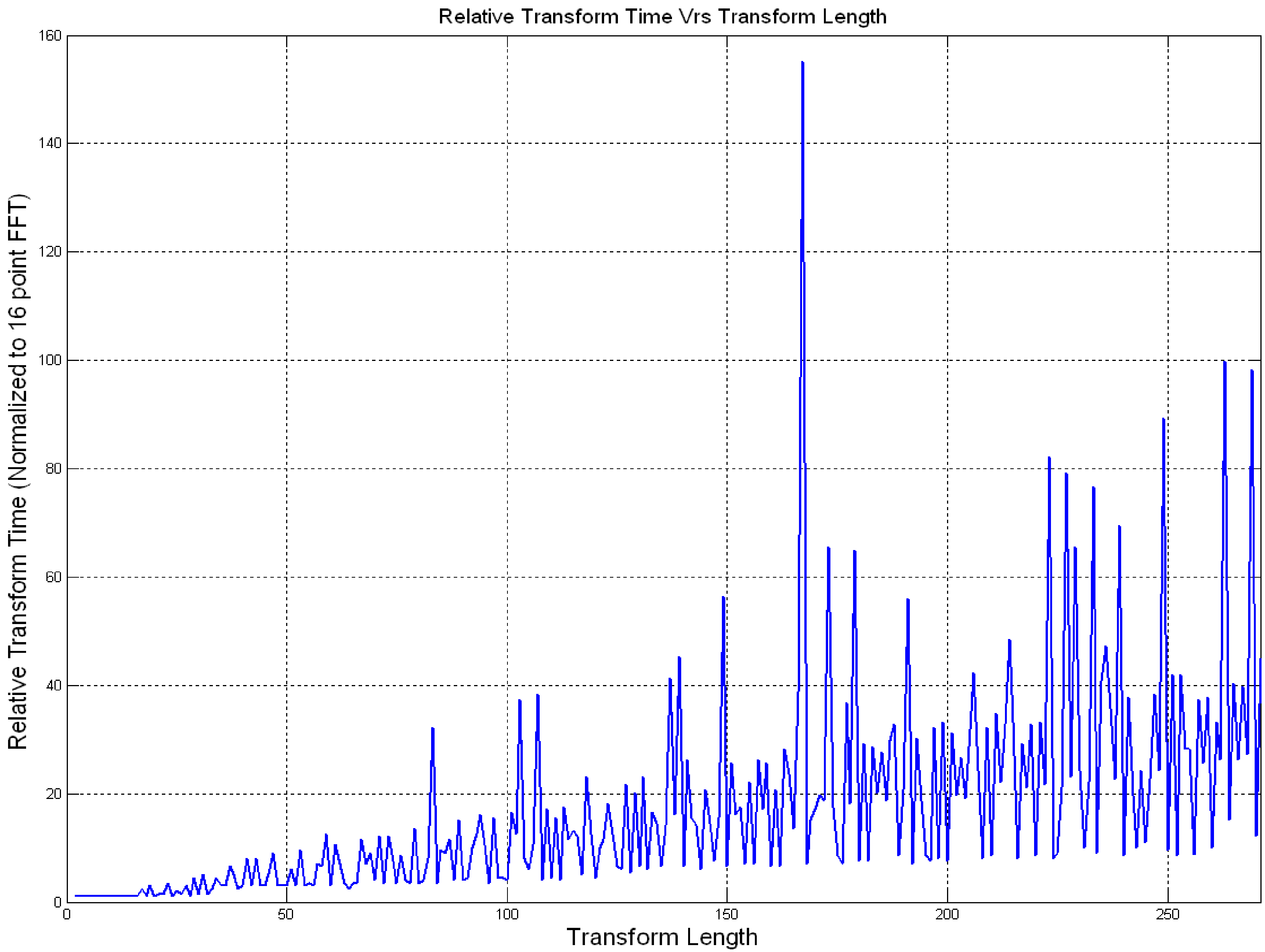
$$W^{-1} = \frac{1}{N} W^{H} = \frac{1}{N} W^{*}$$

A FAST FOURIER TRANSFORM

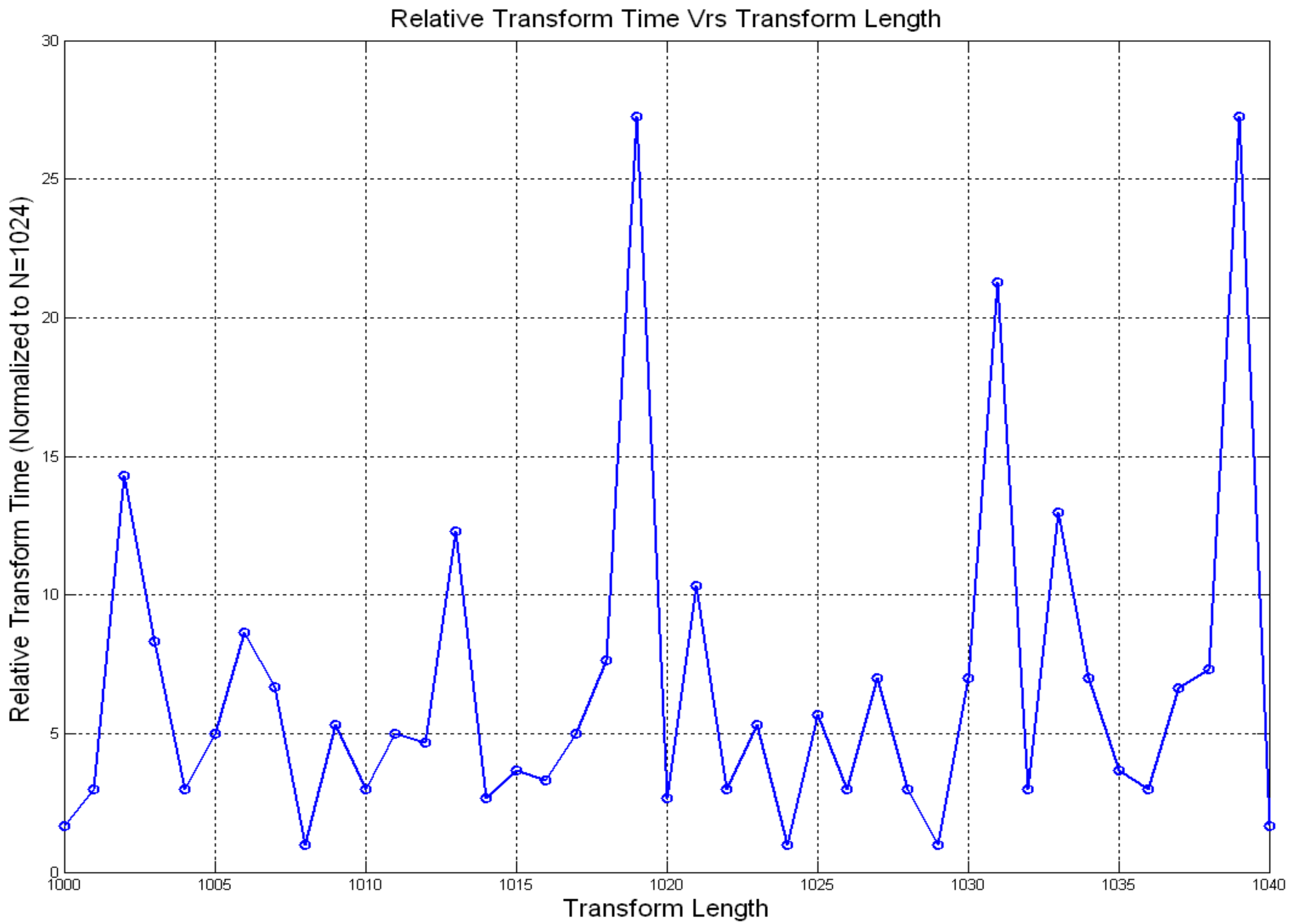


Maps One-Dimensional Array to a Two Dimensional Array,
Performs a 2-D Transform and
Maps Two-Dimensional Array to a One Dimensional Array,

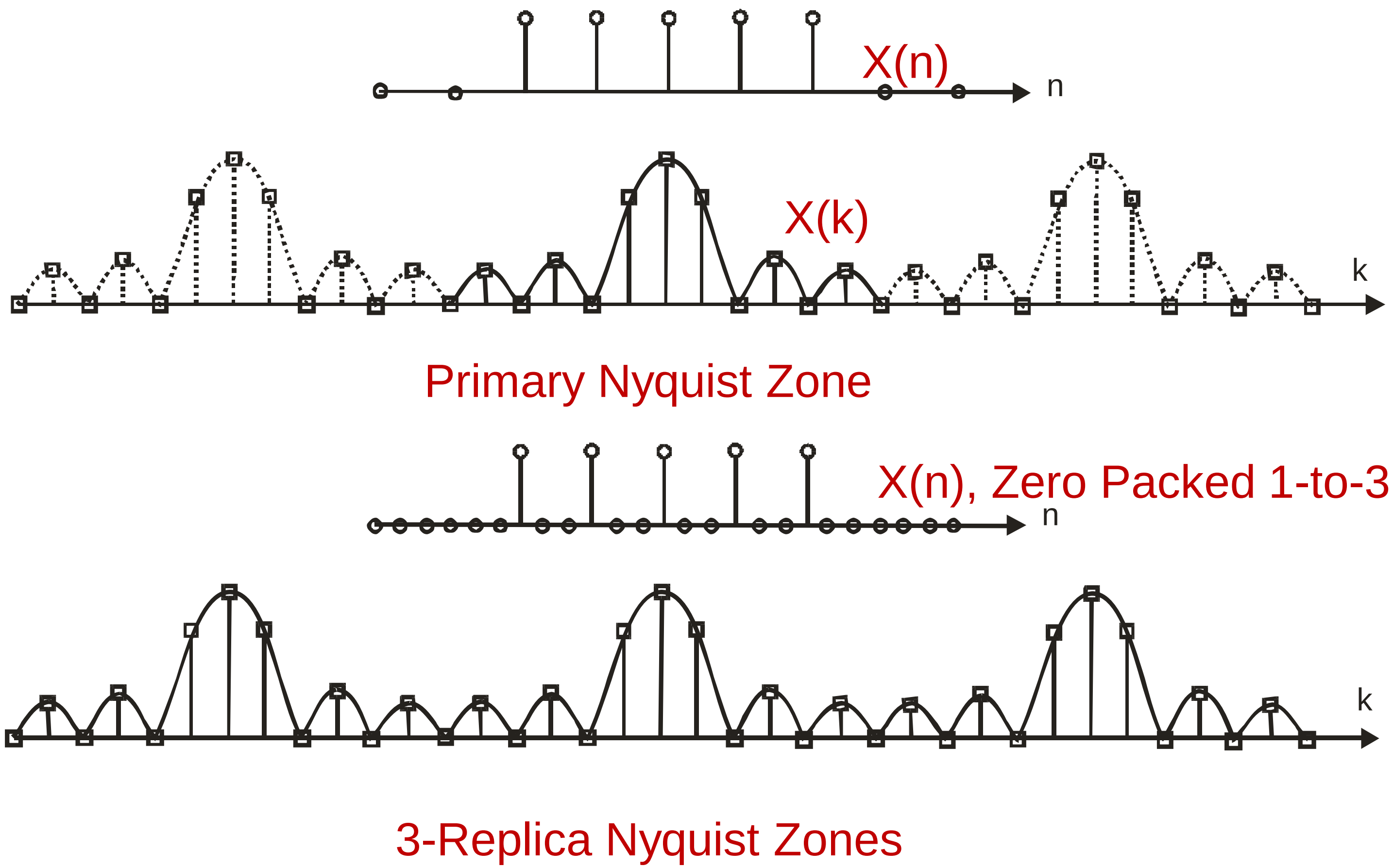
RELATIVE TIME FOR FFTS OF LENGTH 2 TO 270

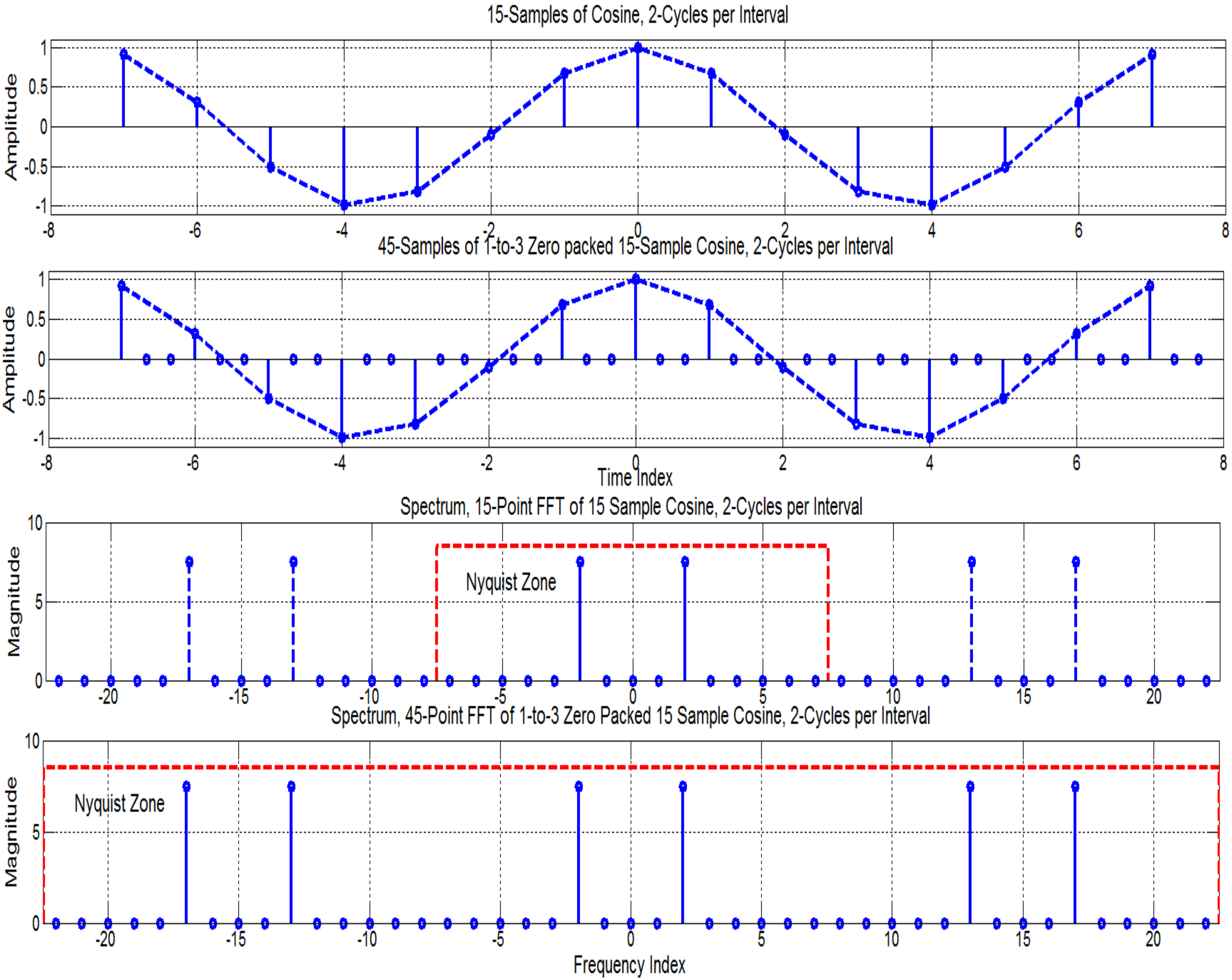


Relative Time for FFTs of Length 1000 to 1040



ZERO-PACKED DATA HAS REPLICA SPECTRA




15-Samples of Cosine, 2-Cycles per Interval
Amplitude
45-Samples of 1-to-3 Zero packed 15-Sample Cosine, 2-Cycles per Interval
Amplitude
Time Index
Spectrum, 15-Point FFT of 15 Sample Cosine, 2-Cycles per Interval
Magnitude
Nyquist Zone
Spectrum, 45-Point FFT of 1-to-3 Zero Packed 15 Sample Cosine, 2-Cycles per Interval
Magnitude
Nyquist Zone
Frequency Index

COOLEY-TUKEY INDEX MAPPING

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

Array Indices

0	-	-	3	-	-	6	-	-	9	-	-	12	-	-
-	1	-	-	4	-	-	7	-	-	10	-	-	13	-
-	-	2	-	-	5	-	-	8	-	-	11	-	-	14

Sieved Array

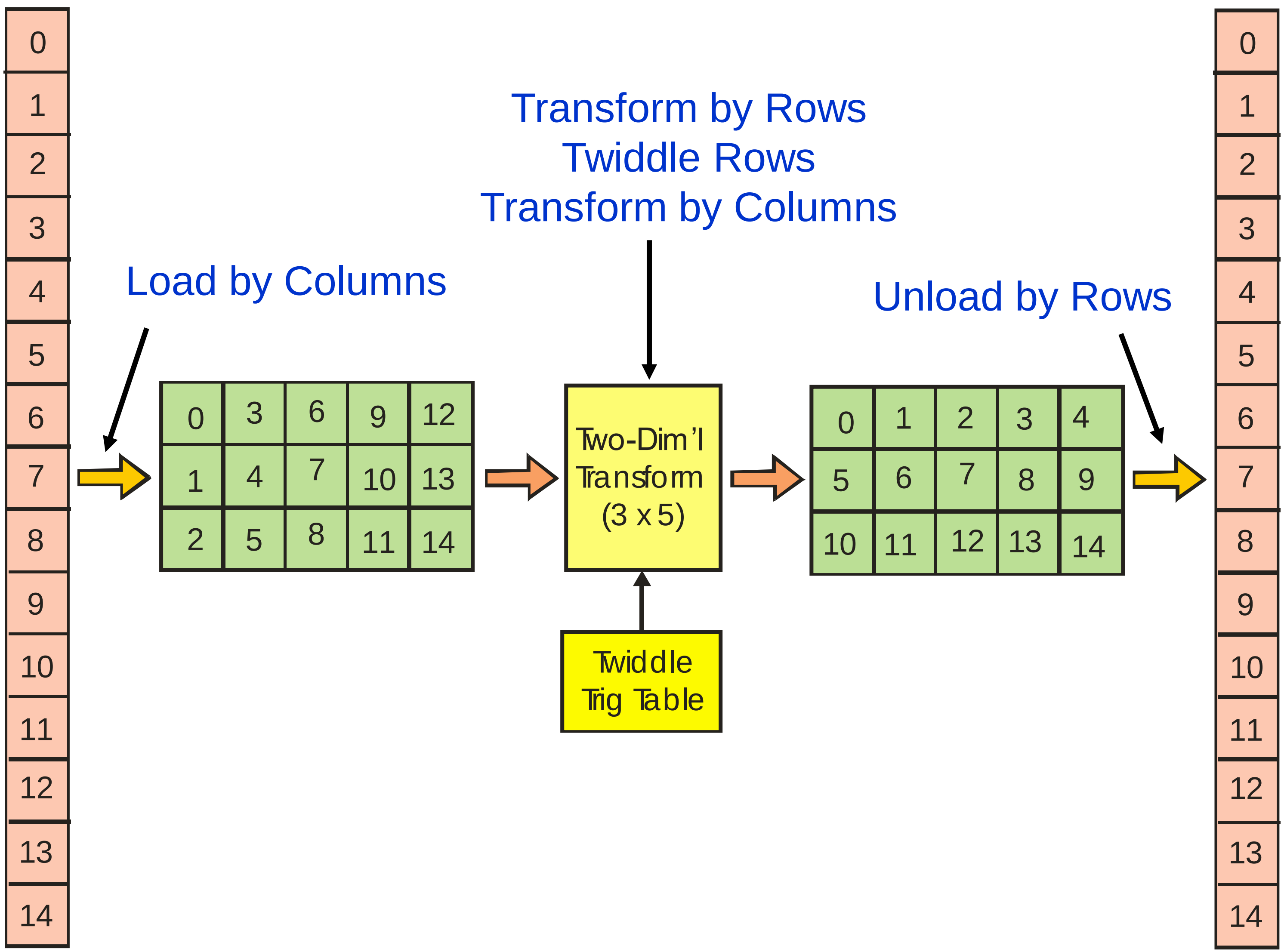
0	3	6	9	12		
-	1	4	7	10	13	
-	-	2	5	8	11	14

Remove Gaps (Contain Empty Addresses)

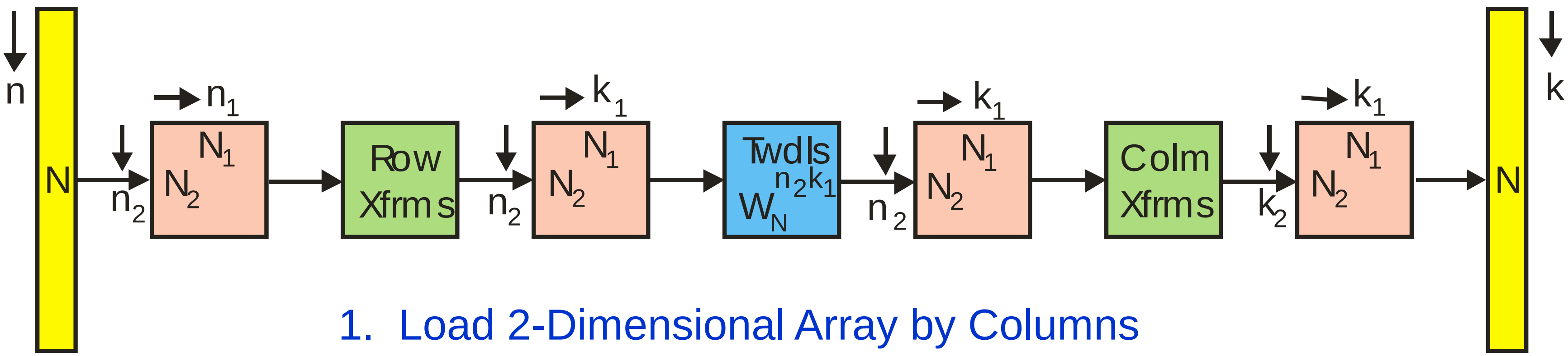
0	3	6	9	12
1	4	7	10	13
2	5	8	11	14

Shift to Common Origin

LEXICOGRAPHIC (IN NATURAL ORDER) MAPPING



COOLEY-TUKEY TRANSFORM



1. Load 2-Dimensional Array by Columns
2. Transform Rows
3. Twiddle each Element in Transformed Row
4. Transform Columns
5. Unload 2-Dimensional Array by Rows

FFT Workload : $N_1(N_2)^2 + N_1N_2 + N_2(N_1)^2 = (N_1N_2)(N_1 + N_2 + 1)$

DFT Workload : $N^2 = (N_1N_2)(N_1N_2)$

Ratio : $\dfrac{(N_1N_2)(N_1 + N_2 + 1)}{(N_1N_2)(N_1N_2)} = \dfrac{(N_1 + N_2 + 1)}{(N_1N_2)}$

RADIX-2 COOLEY-TUKEY FFT

$$H(k)=\sum_{n=0}^{N-1} h(n)\, e^{-j\frac{2\pi}{N}nk}$$

$$H(k)=\sum_{n=0}^{\frac{N}{2}-1} h(2n)\, e^{-j\frac{2\pi}{N}(2n)k}+\sum_{n=0}^{\frac{N}{2}-1} h(2n+1)\, e^{-j\frac{2\pi}{N}(2n+1)k}$$

$$=\sum_{n=0}^{\frac{N}{2}-1} h(2n)\, e^{-j\frac{2\pi}{N/2}nk}+e^{-j\frac{2\pi}{N}k}\sum_{n=0}^{\frac{N}{2}-1} h(2n+1)\, e^{-j\frac{2\pi}{N/2}nk}$$

N/2 Point DFT of Even Indexed Data

Twiddle

N/2 Point DFT of Odd Indexed Data

$$\mathrm{H}(k+\frac{N}{2})=\sum_{n=0}^{\frac{N}{2}-1} h(2n)\, e^{-j\frac{2\pi}{N/2}n(k+\frac{N}{2})}+e^{-j\frac{2\pi}{N}(k+\frac{N}{2})}\sum_{n=0}^{\frac{N}{2}-1} h(2n+1)\, e^{-j\frac{2\pi}{N/2}n(k+\frac{N}{2})}$$

$$=\sum_{n=0}^{\frac{N}{2}-1} h(2n)\, e^{-j\frac{2\pi}{N/2}nk}-e^{-j\frac{2\pi}{N}k}\sum_{n=0}^{\frac{N}{2}-1} h(2n+1)\, e^{-j\frac{2\pi}{N/2}nk}$$

COOLEY-TUKEY FAST FOURIER TRANSFORM

$$F(k) = \sum_{n=0}^{N-1} f(n)\, w_N^{nk}$$

$$F(k_1,k_2) = \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} f(n_1,n_2)\, w_{N_1N_2}^{(n_1+n_2N_1)(k_1+k_2N_2)}$$

Examine product in Exponent

$$w_N^{nk} = w_N^{n_1k_1}\; w_N^{n_1k_2N_2}\; w_N^{n_2k_1N_1}\; \underbrace{w_N^{n_2k_2N_1N_2}}_{N_1N_2=N}$$

$$= w_N^{n_1k_1}\; w_{N_1}^{n_1k_2}\; w_{N_2}^{n_2k_1}$$

$$F(k_1,k_2) = \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} f(n_1,n_2)\, w_N^{n_1k_1}\; w_{N_1}^{n_1k_2}\; w_{N_2}^{n_2k_1}$$

$$= \sum_{n_1=0}^{N_1-1} \left[w_N^{n_1k_1} \sum_{n_2=0}^{N_2-1} f(n_1,n_2)\, w_{N_2}^{n_2k_1} \right] w_{N_1}^{n_1k_2}$$

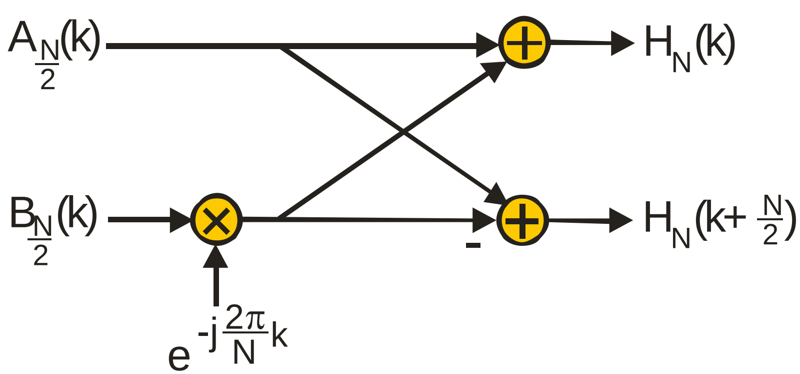
RADIX-2 BUTTERFLY

$$H(k) = \sum_{n=0}^{\frac{N}{2}-1} h(2n)\, e^{-j\frac{2\pi}{N/2}nk} + e^{-j\frac{2\pi}{N}k} \sum_{n=0}^{\frac{N}{2}-1} h(2n+1)\, e^{-j\frac{2\pi}{N/2}nk}$$

$$H(k+\frac{N}{2}) = \sum_{n=0}^{\frac{N}{2}-1} h(2n)\, e^{-j\frac{2\pi}{N/2}nk} - e^{-j\frac{2\pi}{N}k} \sum_{n=0}^{\frac{N}{2}-1} h(2n+1)\, e^{-j\frac{2\pi}{N/2}nk}$$

$$H(k) = A(k) + e^{-j\frac{2\pi}{N}k} B(k)$$

$$H(k+\frac{N}{2}) = A(k) - e^{-j\frac{2\pi}{N}k} B(k)$$



Butterfly
(Artistic License)

IN CASE YOU FORGOT,
IMAGE OF REAL BUTTERFLY!

A Puzzle!

N^2 Multiplies for N-Point DFT

$$H(k)=\sum_{n=0}^{N-1} h(n)\, e^{-j\frac{2\pi}{N}nk}$$

Every Multiply in Original Summation is either in Left Hand in Right hand Summation

$$H(k)=\sum_{n=0}^{\frac{N}{2}-1} h(2n)\, e^{-j\frac{2\pi}{N}(2n)k}+\sum_{n=0}^{\frac{N}{2}-1} h(2n+1)\, e^{-j\frac{2\pi}{N}(2n+1)k}$$

$$=\sum_{n=0}^{\frac{N}{2}-1} h(2n)\, e^{-j\frac{2\pi}{N/2}nk}+e^{-j\frac{2\pi}{N}k}\sum_{n=0}^{\frac{N}{2}-1} h(2n+1)\, e^{-j\frac{2\pi}{N/2}nk}$$

N/2 point DFT of Even Indexed Sample Points

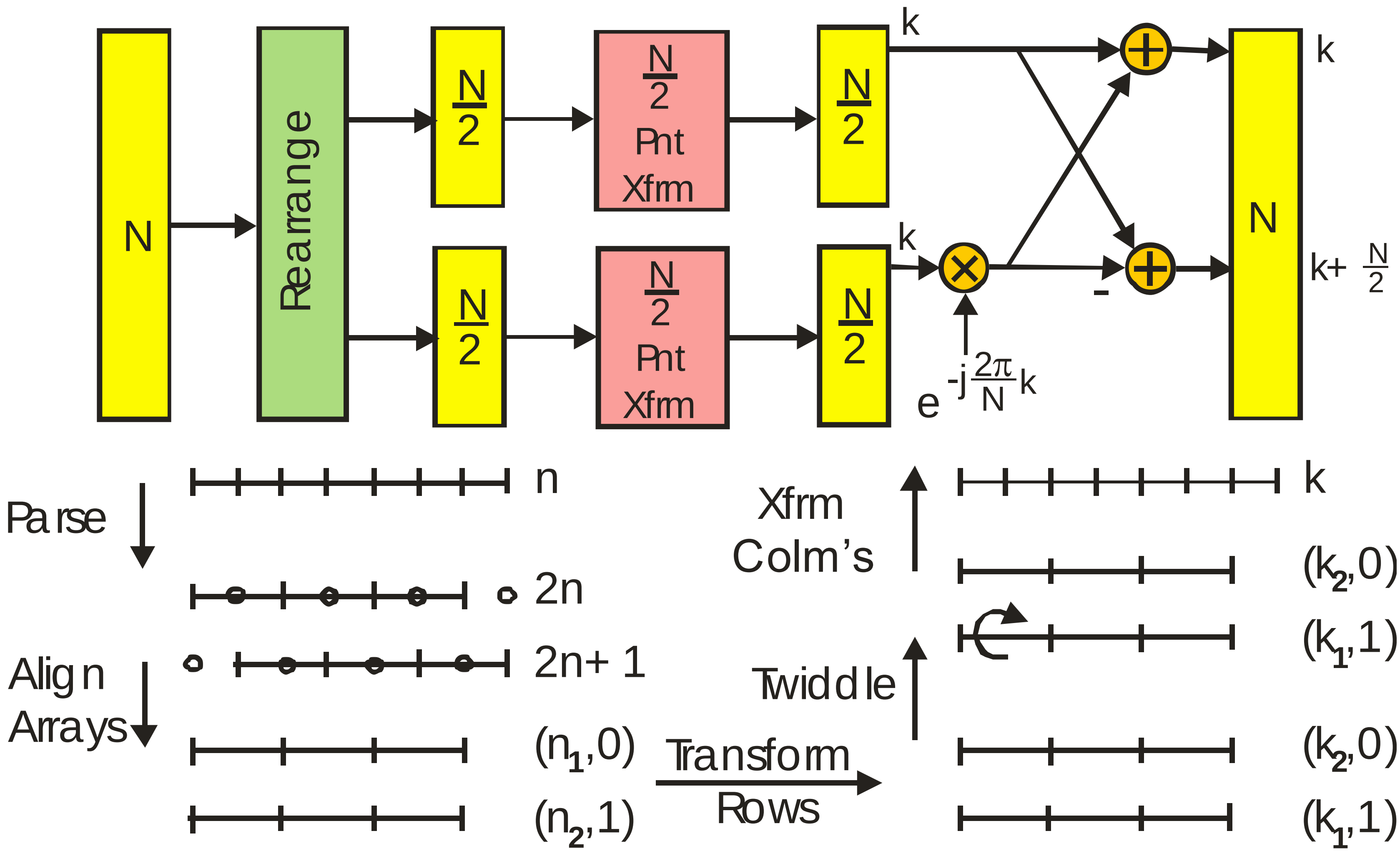
N/2 point DFT of Odd Indexed Sample Points

$(N/2)^2$ Multiplies in (N/2) Point DFT

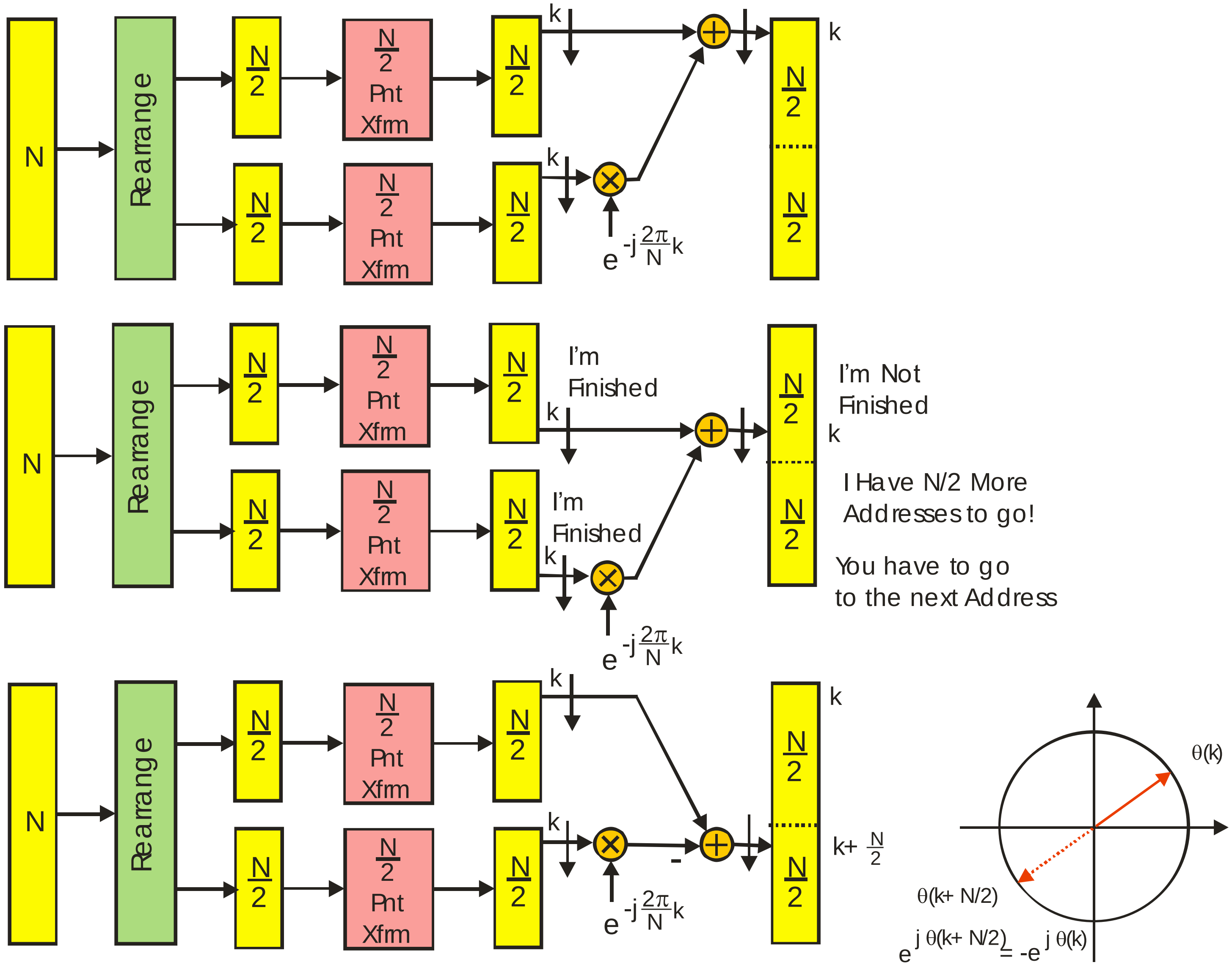
$(N/2)^2+(N/2)^2 = N^2/4 + N^2/4 = N^2/2$

What Happened to Half of the Workload?

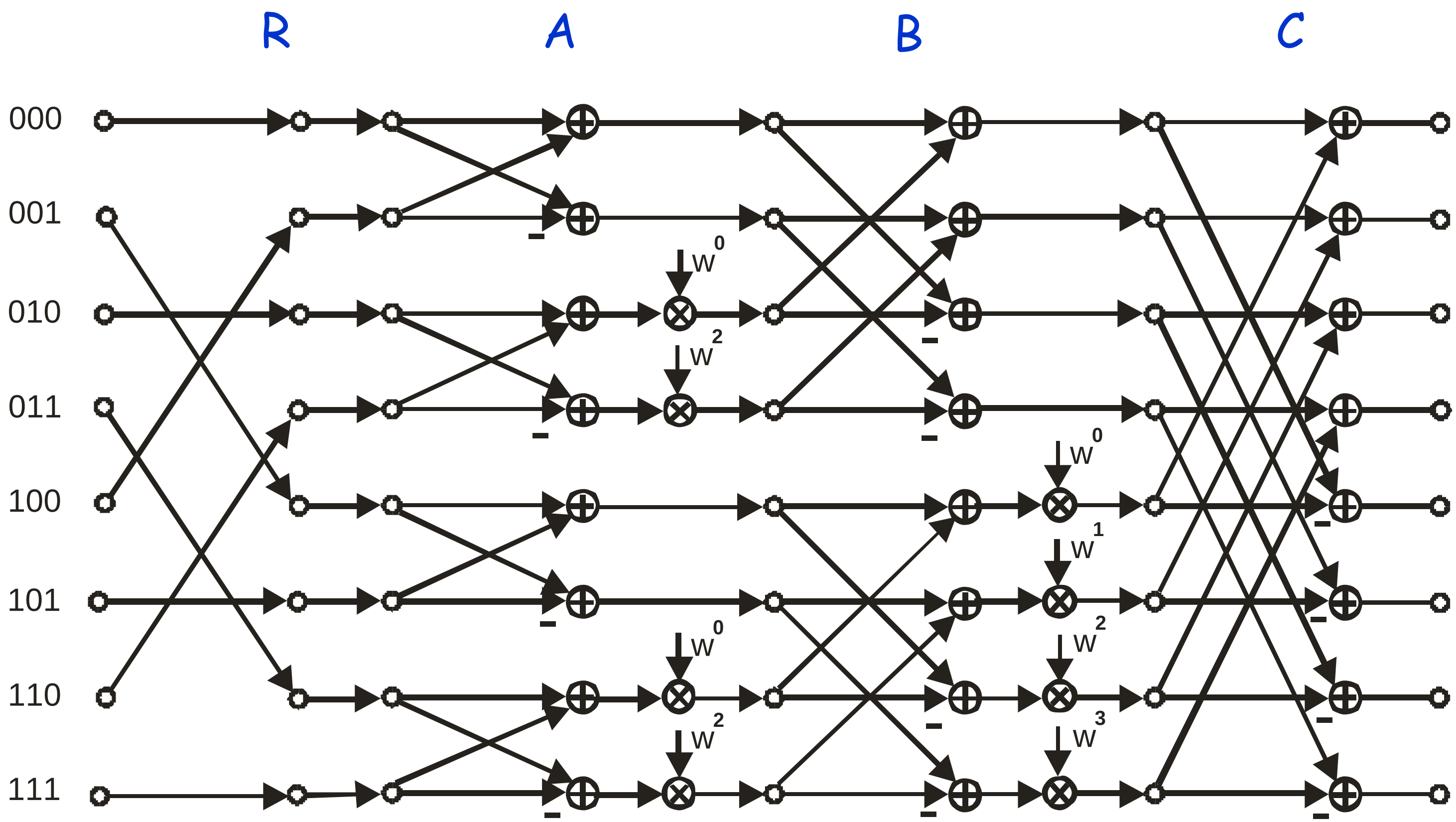
TWIDDLE: A FREQUENCY DEPENDENT PHASE SHIFT TO DELAY SHIFTED TIME SERIES



N/2 INPUT ADDRESSES, N-OUTPUT ADDRESSES

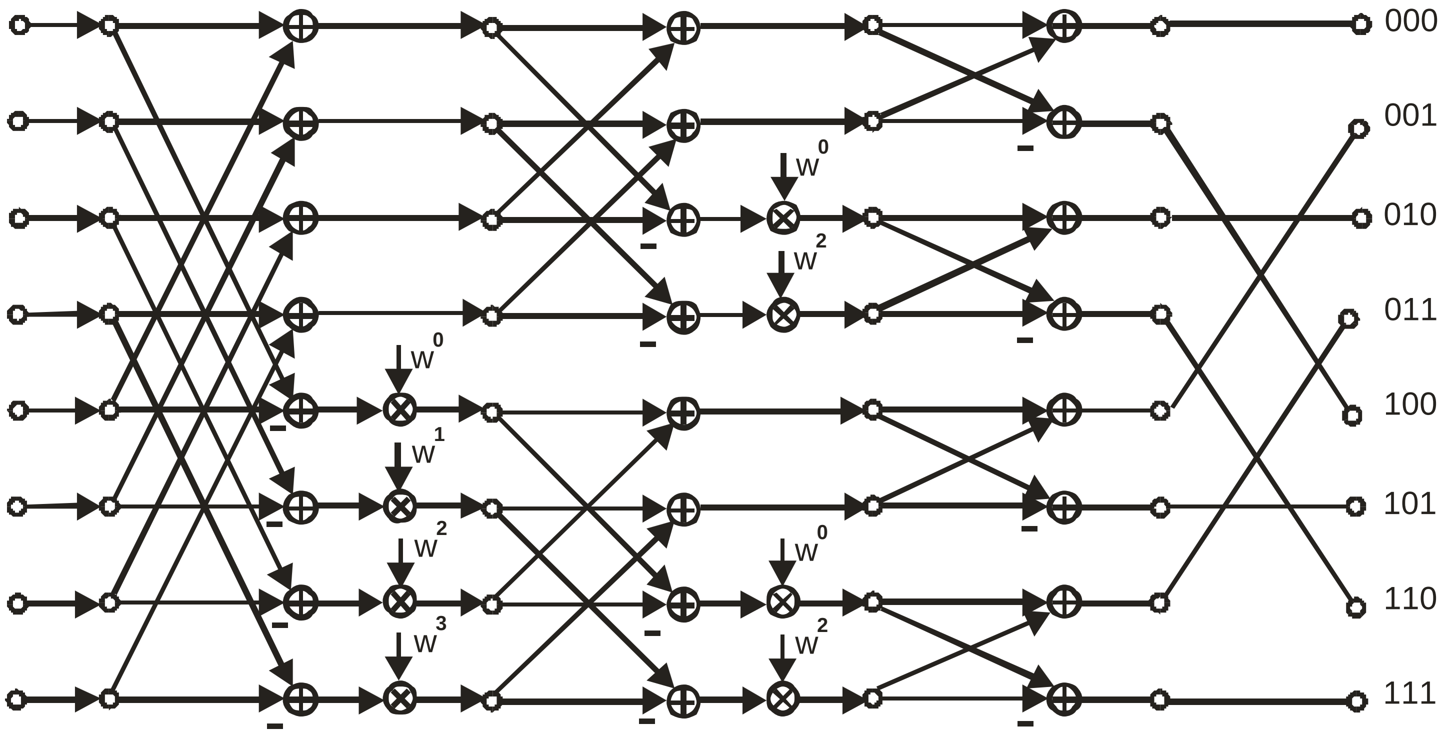


RADIX-2 COOLEY-TUKEY REARRANGEMENT IN TIME



W=C B A R

RADIX-2 COOLEY-TUKEY REARRANGEMENT IN FREQUENCY

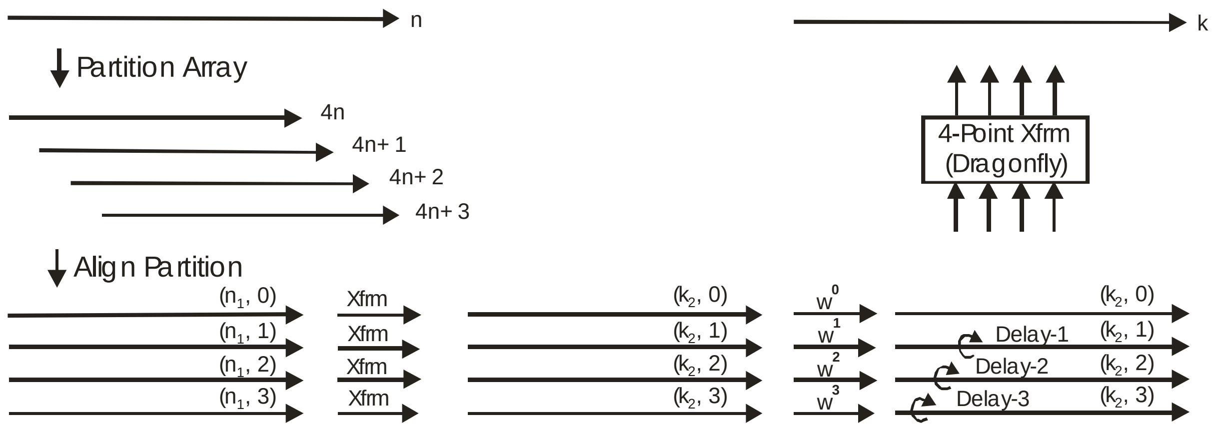


$W = W^T$:   Then, $W^T = [C\ B\ A\ R]^T = R^T A^T B^T C^T = R\ A^T B^T C^T$

Transpose of Product is Product of Transposes In Opposite Order

Also Seen as Dual Graph: Replace Nodes With Sums,
Replace Sums with Nodes, Reverse Direction of Arrows

SIGNAL FLOW GRAPH OF RADIX-4 TRANSFORM AND DRAGONFLY

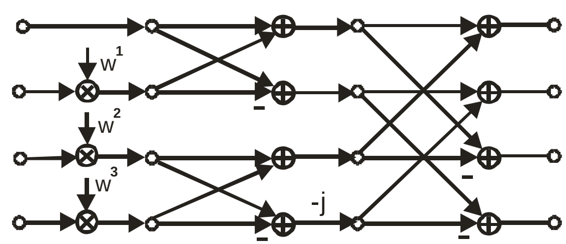


$$F(k) = A_0(k) + W^k A_1(k) + W^{2k} A_2(k) + W^{3k} A_3(k)$$

$$F(k+\frac{N}{4}) = A_0(k) - j\,W^k A_1(k) - W^{2k} A_2(k) + j\,W^{3k} A_3(k)$$

$$F(k+2\frac{N}{4}) = A_0(k) - W^k A_1(k) + W^{2k} A_2(k) - W^{3k} A_3(k)$$

$$F(k+3\frac{N}{4}) = A_0(k) + j\,W^k A_1(k) - W^{2k} A_2(k) - j\,W^{3k} A_3(k)$$

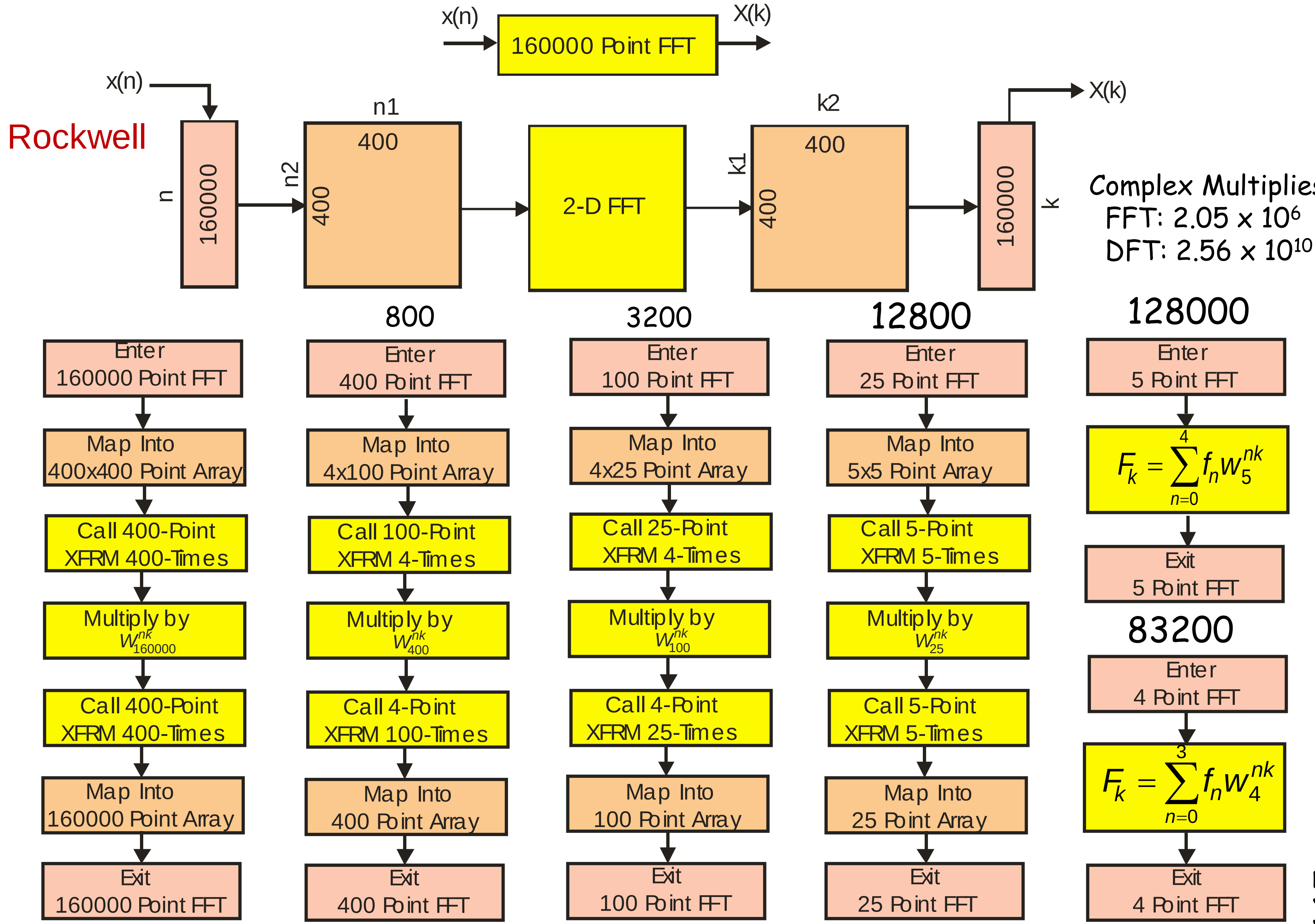


Dragonfly

1-Dragonfly -> 4-Butterflies
25% Reduction in Twiddles
50% Reduction in Memory Calls

Image of Real Dragonfly

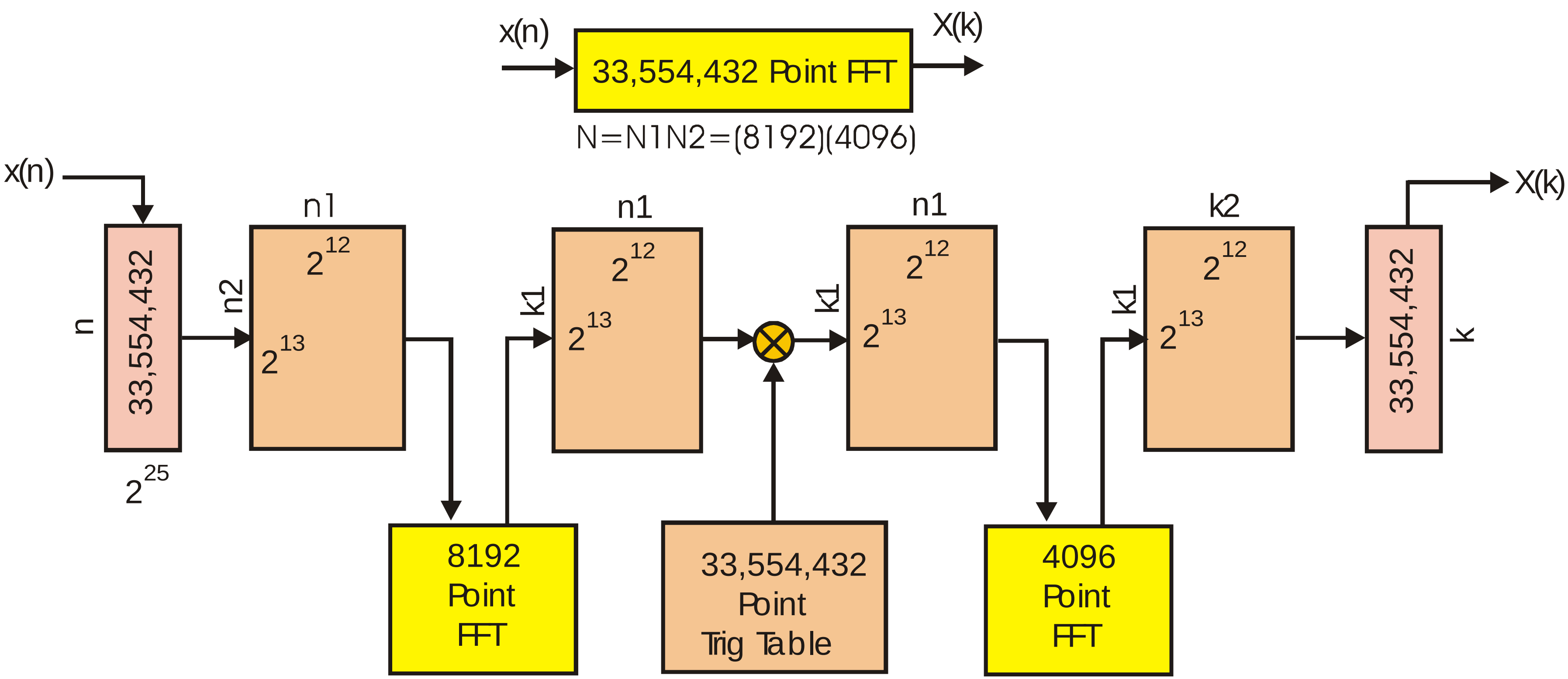
COOLEY-TUKEY MIXED RADIX FFT



$2^{25}=(2^{13})(2^{12})=33{,}554{,}432$ POINT FFT

AMDAHL
Jodrell Bank

Minor Problem:
It Didn't Work!



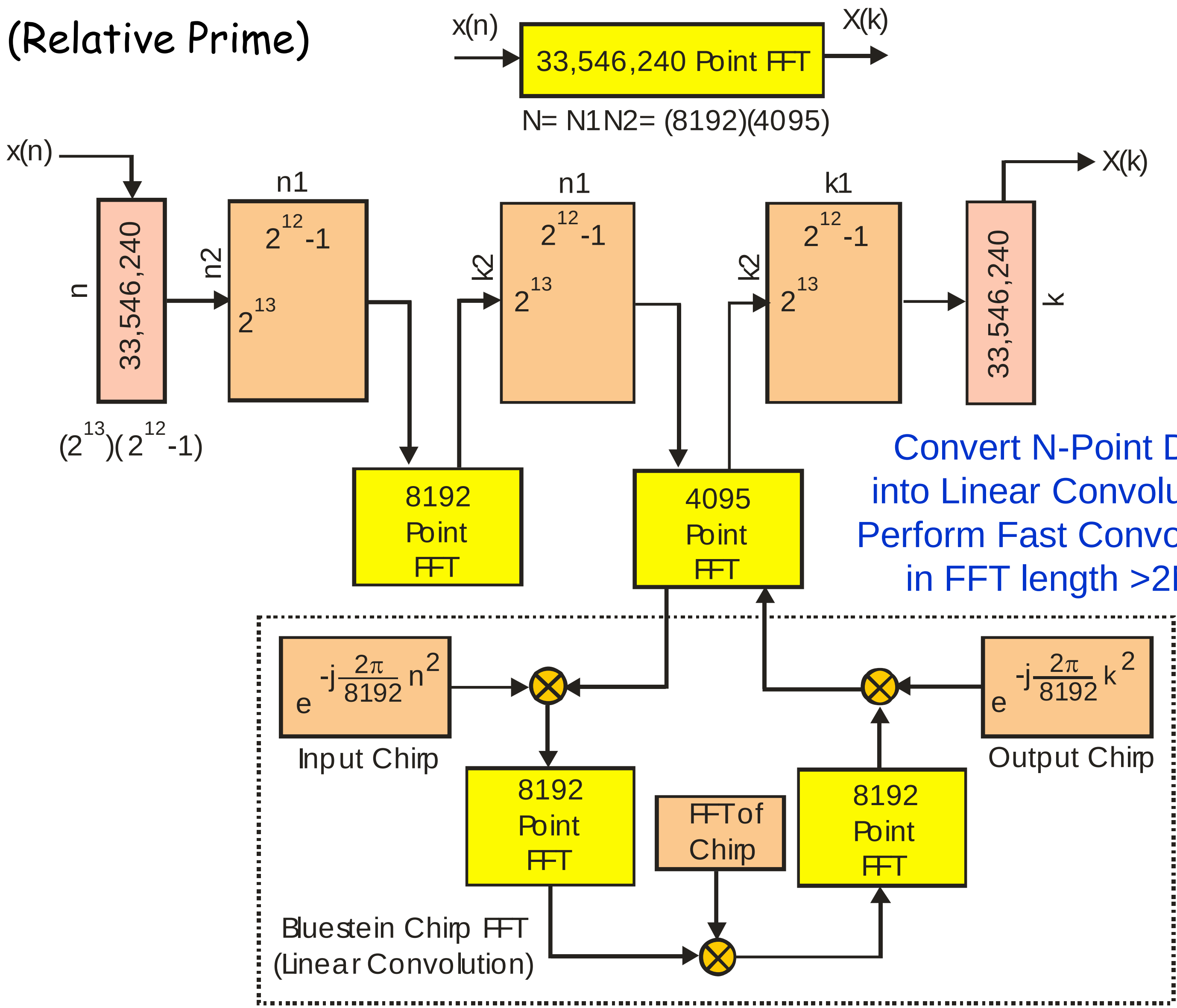
SINE WAVE, ONE CYCLE IN TRIG TABLE

Single precision
Floating Point
24-bit Mantissa
8-bit Exponent

Single Precision Floating Word Has 24-bit precision
Exponent does not Contribute to Arithmetic Precision
Sinewave samples underflow when exponents are aligned for summations

$(2^{13})(2^{12}-1)=(33,546,240)$ POINT FFT

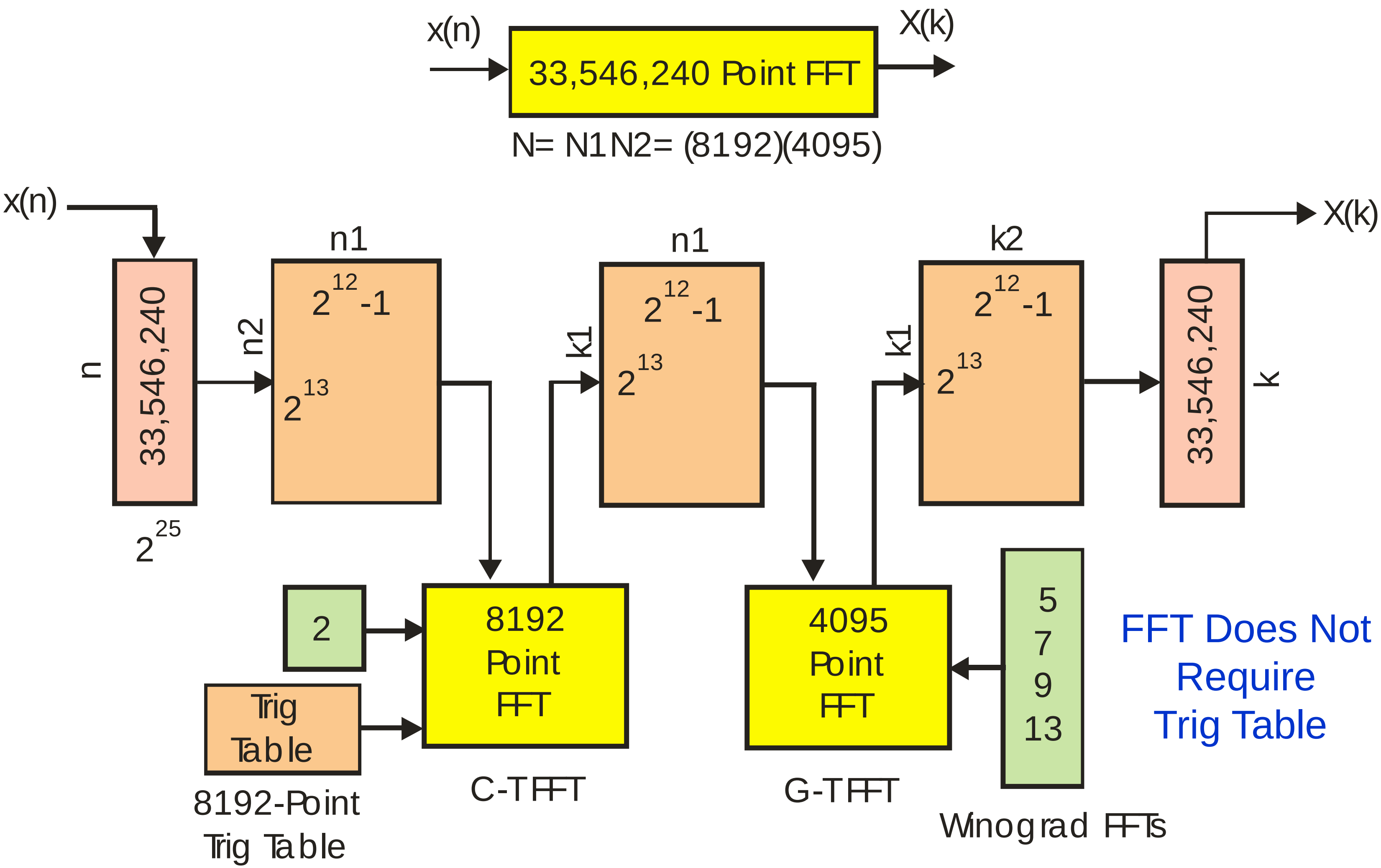
(Relative Prime)



Convert N-Point DFT
into Linear Convolution,
Perform Fast Convolution
in FFT length >2N-1

Requires
2^{13} Point
Trig Table
Instead of
2^{25} Point
Trig Table

$(2^{13})(2^{12}-1)=(33{,}546{,}240)$ POINT FFT

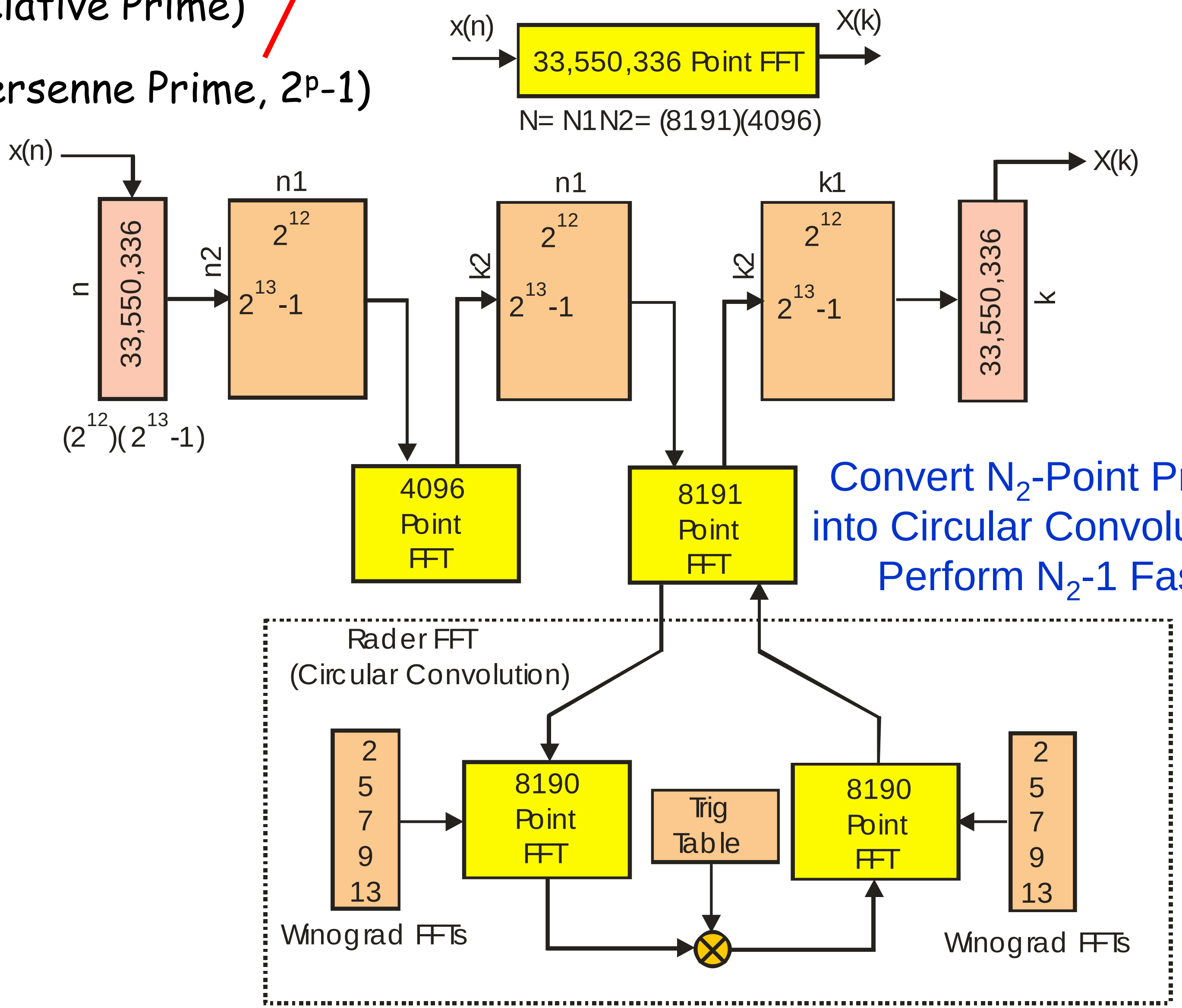


FFT Does Not Require Trig Table

$(2^{12})(2^{13}-1)=(33{,}550{,}336)$ POINT FFT

(Relative Prime)

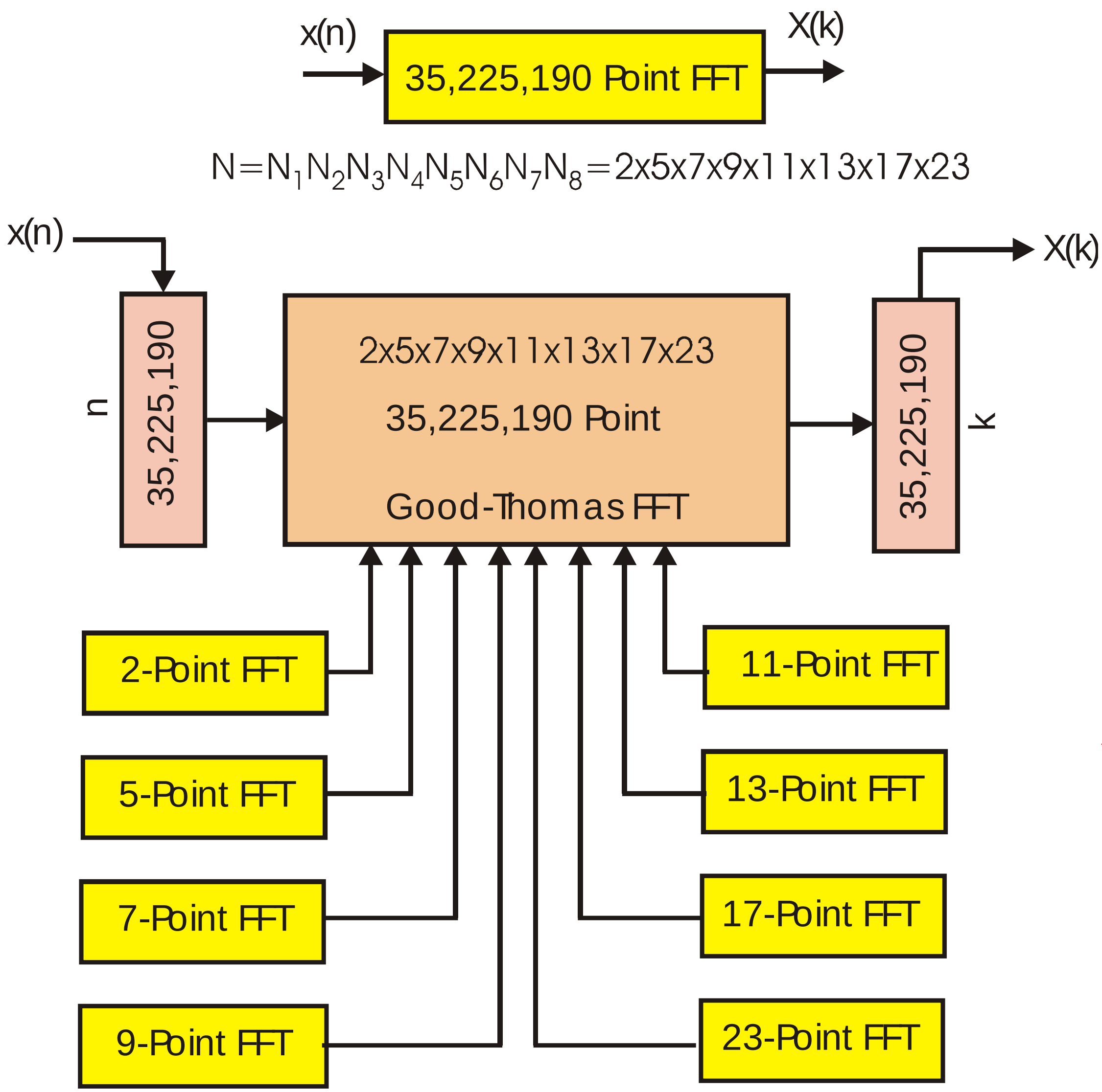
(Mersenne Prime, 2^p-1)



Convert N_2-Point Prime Length DFT into Circular Convolution, Length N_2-1 Perform N_2-1 Fast Convolution

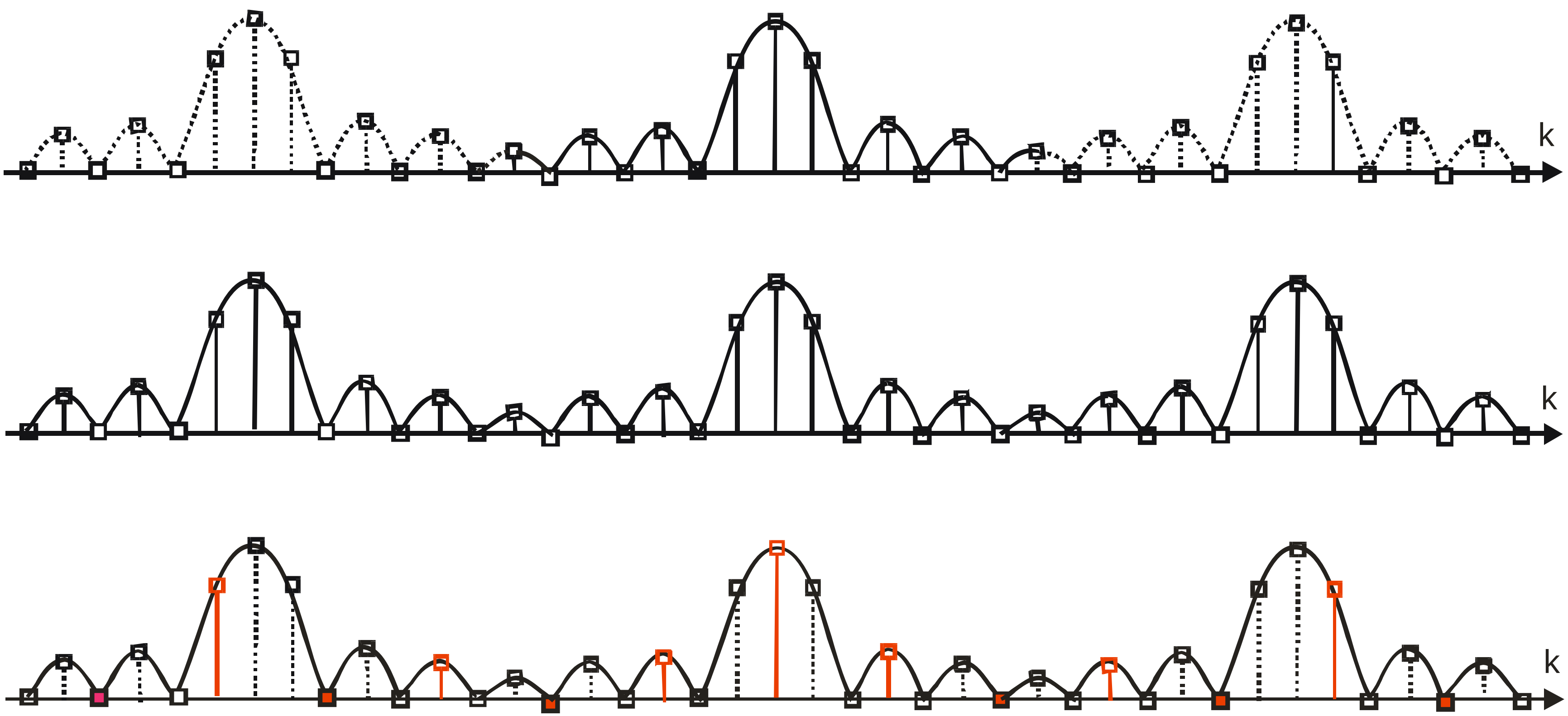
FFT Does Not Require Trig Table

35,225,190 POINT FFT



Short FFT
Algorithms
Trig Table
87 Values

REPLICA SPECTRA OF ZERO-PACKED DATA: DOWN-SAMPLE SPECTRUM AND ALIAS TIME SERIES



Earlier We Suppressed the Spectral Copies by
3-to-1 Down-Sample of the Time Series
Here We take a Different Tack:
We 3-to-1 Down-Sample the Spectral Copies!

GOOD-THOMAS INDEX MAPPING

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

Array Indices

0	-	-	3	-	-	6	-	-	9	-	-	12	-	-
-	1	-	-	4	-	-	7	-	-	10	-	-	13	-
-	-	2	-	-	5	-	-	8	-	-	11	-	-	14

Sieved Array

0	-	-	3	-	-	6	-	-	9	-	-	12	-	-
-	1	-	-	4	-	-	7	-	-	10	-	-	13	-
-	-	2	-	-	5	-	-	8	-	-	11	-	-	14

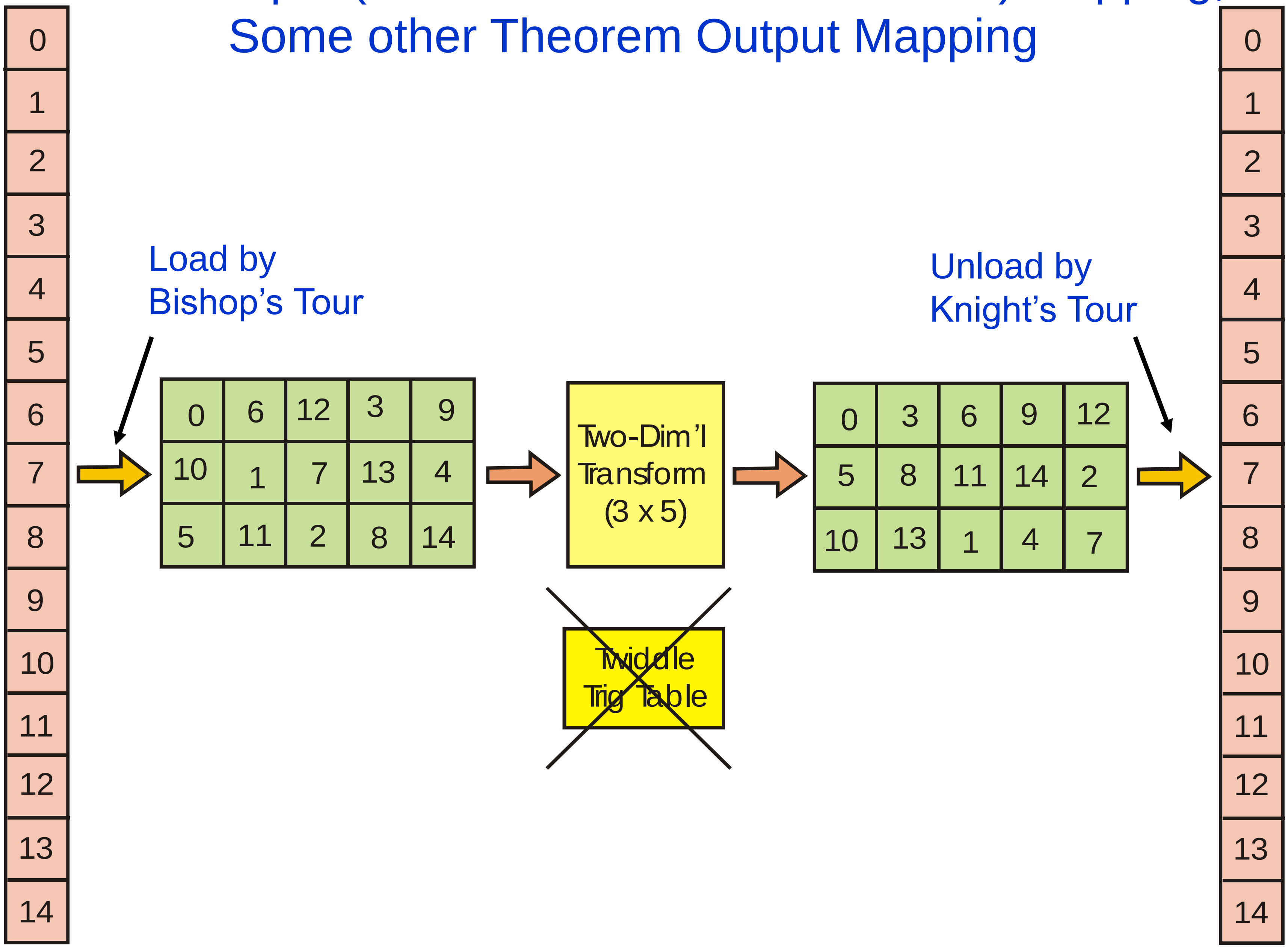
Aliasing Boundaries Due to Down - Sampling Spectrum

0	6	12	3	9
10	1	7	13	4
5	11	2	8	14

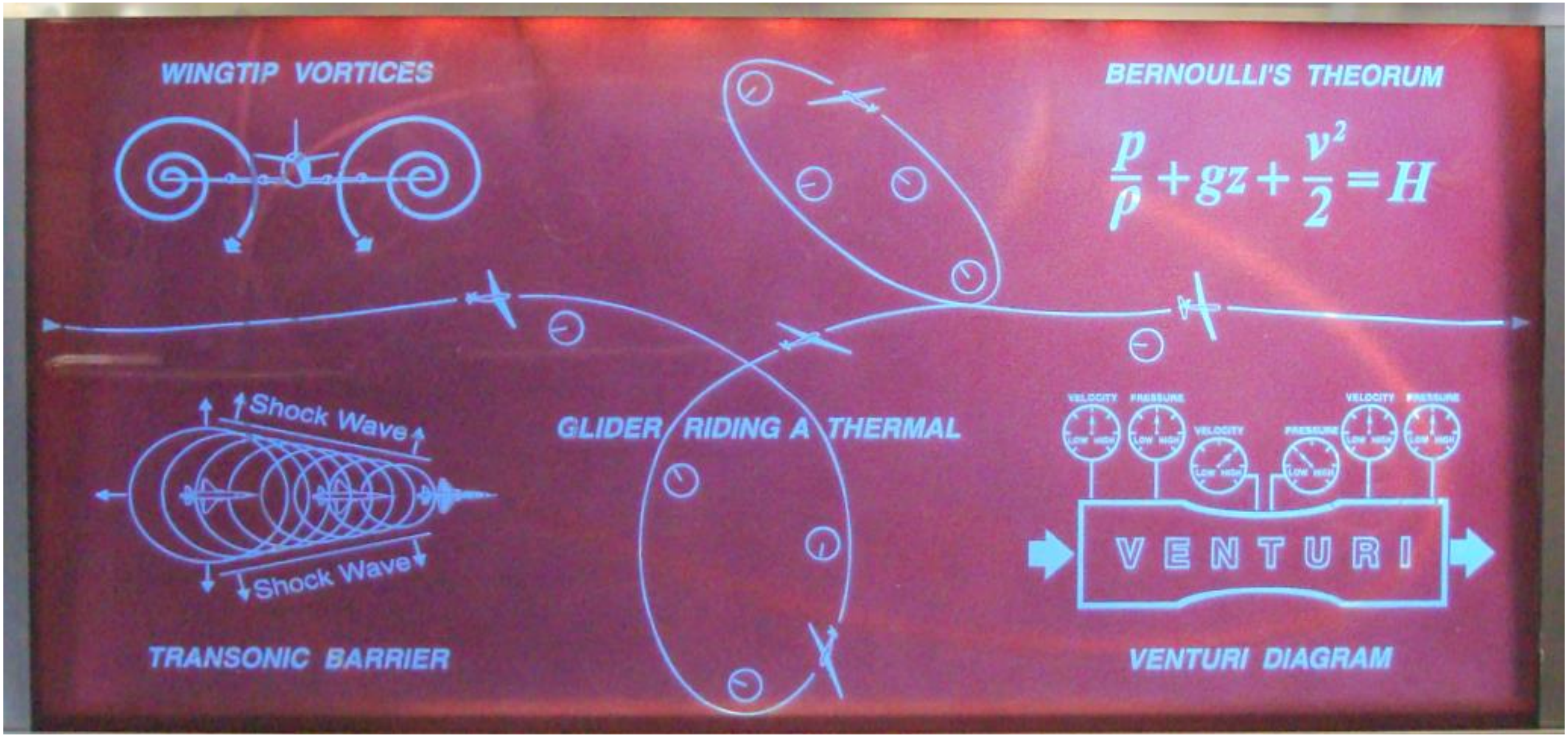
Alaised Time Series Due to Down - Sampling Spectrum

Good-Thomas
As opposed to
Odd-Thomas

Residue Input (Chinese Remainder Theorem) Mapping, Some other Theorem Output Mapping



Speaking of "Theorem"



We all have need for a spell checker!

```
x1 = [ 0   1   2   3   4   5   6   7   -7   -6   -5   -4   -3   -2   -1]        % Odd Symmetric Input
fx1=imag(fft(x1))                                                                % Transform Input Array

fx1=[ 0  -36.0730   18.4395  -12.7598   10.0922   -8.6603    7.8860   -7.5413
           7.5413   -7.8860    8.6603  -10.0922   12.7598  -18.4395   36.0730]

x2=[x1(1)   x1(7)   x1(13)  x1(4)   x1(10);                                      % Map 1-D array to 2-D array
    x1(11)  x1(2)   x1(8)   x1(14)  x1(5) ;
    x1(6)   x1(12)  x1(3)   x1(9)   x1(15)]

x2 =    0    6   -3    3   -6
       -5    1    7   -2    4
        5   -4    2   -7   -1

fx2=[fft(x2(1,:));fft(x2(2,:));fft(x2(3,:))]                                     % Transform Rows

fx2 =   0.0000 + 0.0000i   0.0000 - 7.8860i   0.0000 -12.7598i   0.0000 +12.7598i   0.0000 + 7.8860i
        5.0000 + 0.0000i  -7.5000 - 2.4369i  -7.5000 +10.3229i  -7.5000 -10.3229i  -7.5000 + 2.4369i
       -5.0000 + 0.0000i   7.5000 - 2.4369i   7.5000 +10.3229i   7.5000 -10.3229i   7.5000 + 2.4369i

fx3=[fft(fx2(:,1)) fft(fx2(:,2)) fft(fx2(:,3)) fft(fx2(:,4)) fft(fx2(:,5))]% Transform Columns

fx3 =   0.0000 + 0.0000i   0.0000 -12.7598i   0.0000 + 7.8860i   0.0000 - 7.8860i   0.0000 +12.7598i
        0.0000 - 8.6603i   0.0000 + 7.5413i   0.0000 -10.0922i   0.0000 +36.0730i   0.0000 +18.4395i
        0.0000 + 8.6603i   0.0000 -18.4395i   0.0000 -36.0730i   0.0000 +10.0922i   0.0000 - 7.5413i

fx4=imag[fx3(1,1)  fx3(3,3)  fx3(2,5)  fx3(1,2)  fx3(3,4)  fx3(2,1)  fx3(1,3)  fx3(3,5) ...   % Map      2-D
                   fx3(2,2)  fx3(1,4)  fx3(3,1)  fx3(2,3)  fx3(1,5)  fx3(3,2)  fx3(2,4)])    % array to 1-D

fx4=[0  -36.0730   18.4395  -12.7598   10.0922   -8.6603    7.8860   -7.5413
          7.5413   -7.8860    8.6603  -10.0922   12.7598  -18.4395   36.0730]
```

GOOD-THOMAS (RELATIVE PRIME) TRANSFORM

$$F(k)=\sum_{n=0}^{N-1} f(n)\, w_N^{nk} \qquad w_N^{nk}=e^{-j\frac{2\pi}{N}nk}$$

$$n=0,1,2,\ldots,N-1$$

$$k=0,1,2,\ldots,N-1$$

$$N=N_1N_2 \qquad GCD(N_1,N_2)=1$$

2 *3 - 1 *5 = 1

Chinese Remainder Theorem

$$\begin{cases} M_1N_1+M_2N_2=1 \\ n=[n_1M_2N_2+n_2M_1N_1]\bmod(N) \\ n_1=n\ Mod(N_1),\ n_1=0,1,2,\ldots,N_1-1 \\ n_2=n\ Mod(N_2),\ n_2=0,1,2,\ldots,N_2-1 \end{cases}$$

Ruritanian Correspondence

$$\begin{cases} k=[k_1N_2+k_2N_1]\bmod(N) \\ k_1=kM_1\bmod(N_1) \\ k_2=kM_2\bmod(N_2) \end{cases}$$

GOOD-THOMAS FAST FOURIER TRANSFORM

$$F(k)=\sum_{n=0}^{N-1} f(n)\, w_N^{nk}$$

$$F(k_1,k_2)=\sum_{n_1=0}^{N_1-1}\sum_{n_2=0}^{N_2-1} f(n_1,n_2)\, w_{N_1N_2}^{(n_1M_2N_2+n_2M_1N_1)(k_1N_2+k_2N_1)}$$

Examine Product in Exponent

$$w_N^{nk}=\underbrace{w_N^{n_1k_1M_2N_2N_2}}_{M_2N_2=(1-M_1N_1)}\ \underbrace{w_N^{n_2k_2M_1N_1N_1}}_{M_1N_1=(1-M_2N_2)}\ \underbrace{w_N^{n_1k_2M_2N_1N_2}}_{N_1N_2=N}\ \underbrace{w_N^{n_2k_1M_1N_1N_2}}_{N_1N_2=N}$$

$$w_N^{nk}=w_N^{(n_1k_1N_2)}\ \underbrace{w_N^{(-n_1k_1M_1N_1N_2)}}_{N_1N_2=N}\ w_N^{(n_2k_2N_1)}\ \underbrace{w_N^{(-n_2k_2M_2N_2N_1)}}_{N_1N_2=N}$$

$$w_N^{nk}=w_{N_1}^{n_1k_1}\quad w_{N_2}^{n_2k_2}$$

$$F(k_1,k_2)=\sum_{n_1=0}^{N_1-1}\sum_{n_2=0}^{N_2-1} f(n_1,n_2)\, w_{N_1}^{n_1k_1}\ w_{N_2}^{n_2k_2}$$

The Math Forum @ Drexel

While we're on the subject of the CRT. There is a similar theorem that seems to be a dual form of the CRT, that I've often seen referred to as the "Ruritanian Correspondence Principle", what is the history of this name?

Discrete and Continuous Fourier Transforms: Analysis, Applications, and Fast Algorithms

The Ruritanian correspondence proposed by Good [24]....

Wikipedia, Prime-factor FFT algorithm

Good's 1958 work on the PFA was cited by Cooley and Tukey (1965). It was the only prior FFT work cited by them.

The Relationship Between Two Fast Fourier Transforms

I. J. Good, IEEE trans on Computers, March 1971

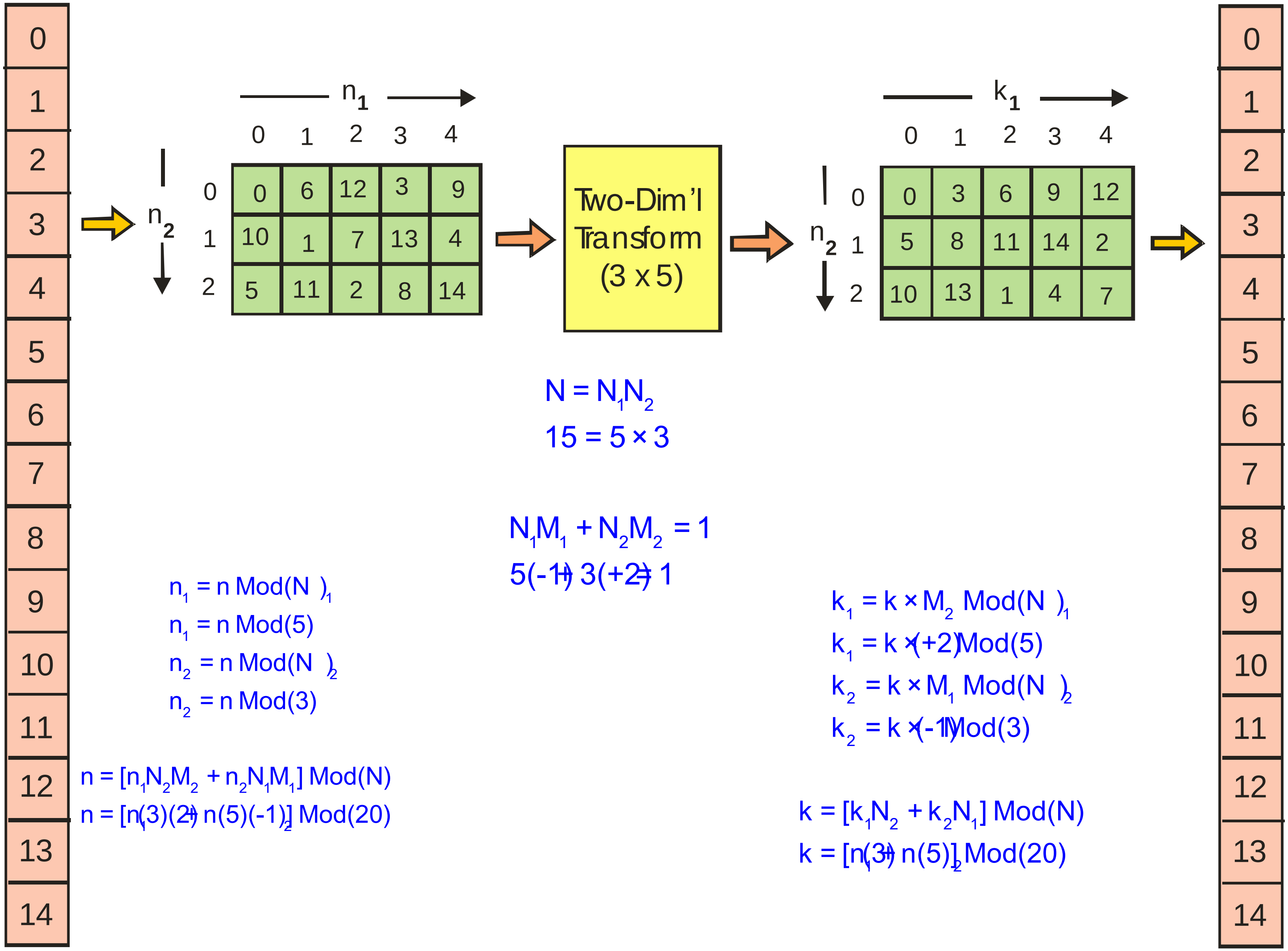
Discussing an equation from his 1958 paper, he added the comment:

.. ... which I shall call the Ruritanian Correspondence.

The interaction algorithm and practical Fourier analysis,

Good, I. J. (1958). *Journal of the Royal Statistical Society,*

CHINESE REMAINDER THEOREM INPUT AND RURITANIAN CORRESPONDENCE OUTPUT MAPPINGS



$$N = N_1 N_2$$
$$15 = 5 \times 3$$

$$N_1 M_1 + N_2 M_2 = 1$$
$$5(-1) + 3(+2) = 1$$

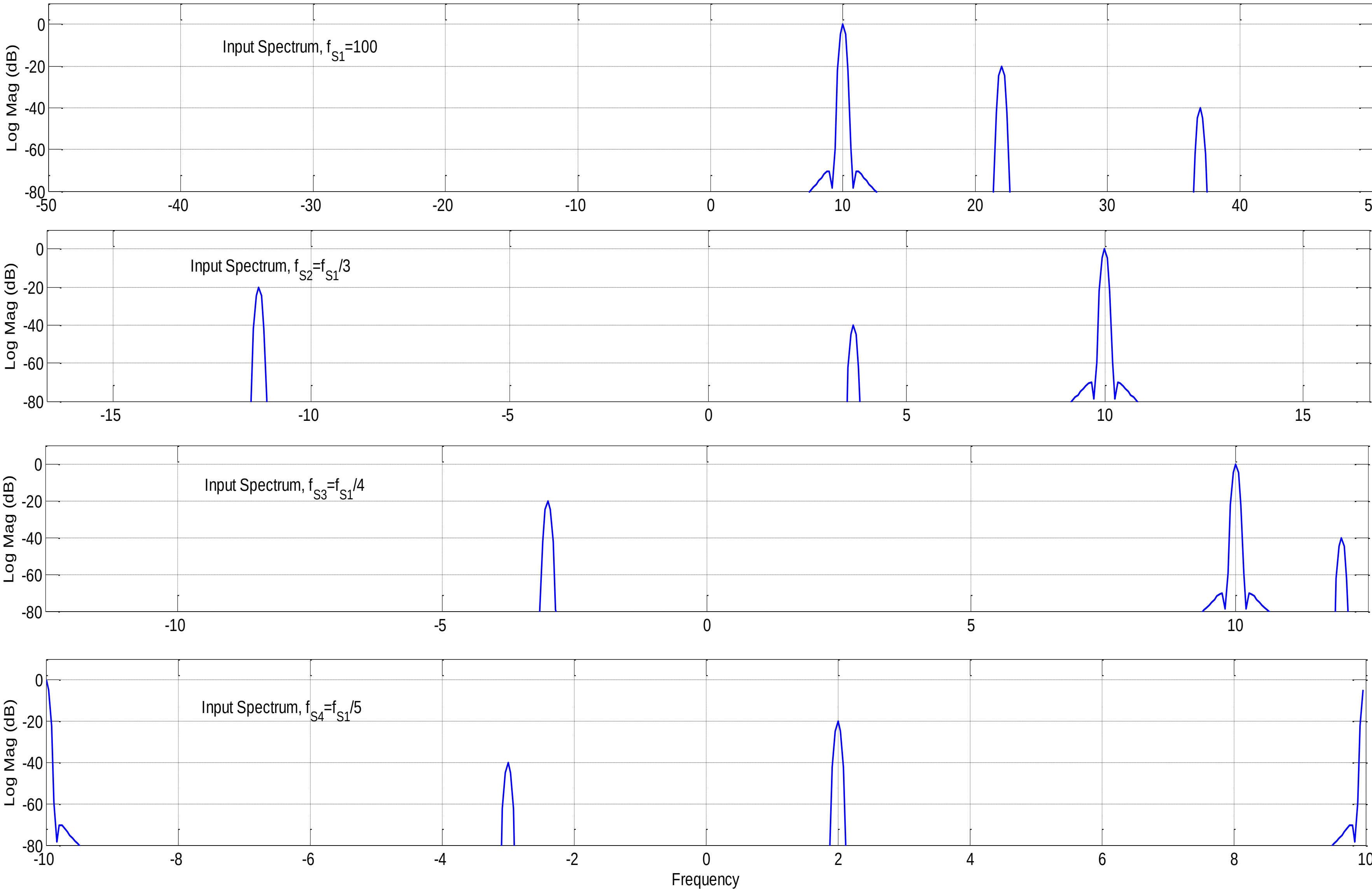
$$n_1 = n \text{ Mod}(N_1)$$
$$n_1 = n \text{ Mod}(5)$$
$$n_2 = n \text{ Mod}(N_2)$$
$$n_2 = n \text{ Mod}(3)$$

$$k_1 = k \times M_2 \text{ Mod}(N_1)$$
$$k_1 = k \times (+2) \text{ Mod}(5)$$
$$k_2 = k \times M_1 \text{ Mod}(N_2)$$
$$k_2 = k \times (-1) \text{ Mod}(3)$$

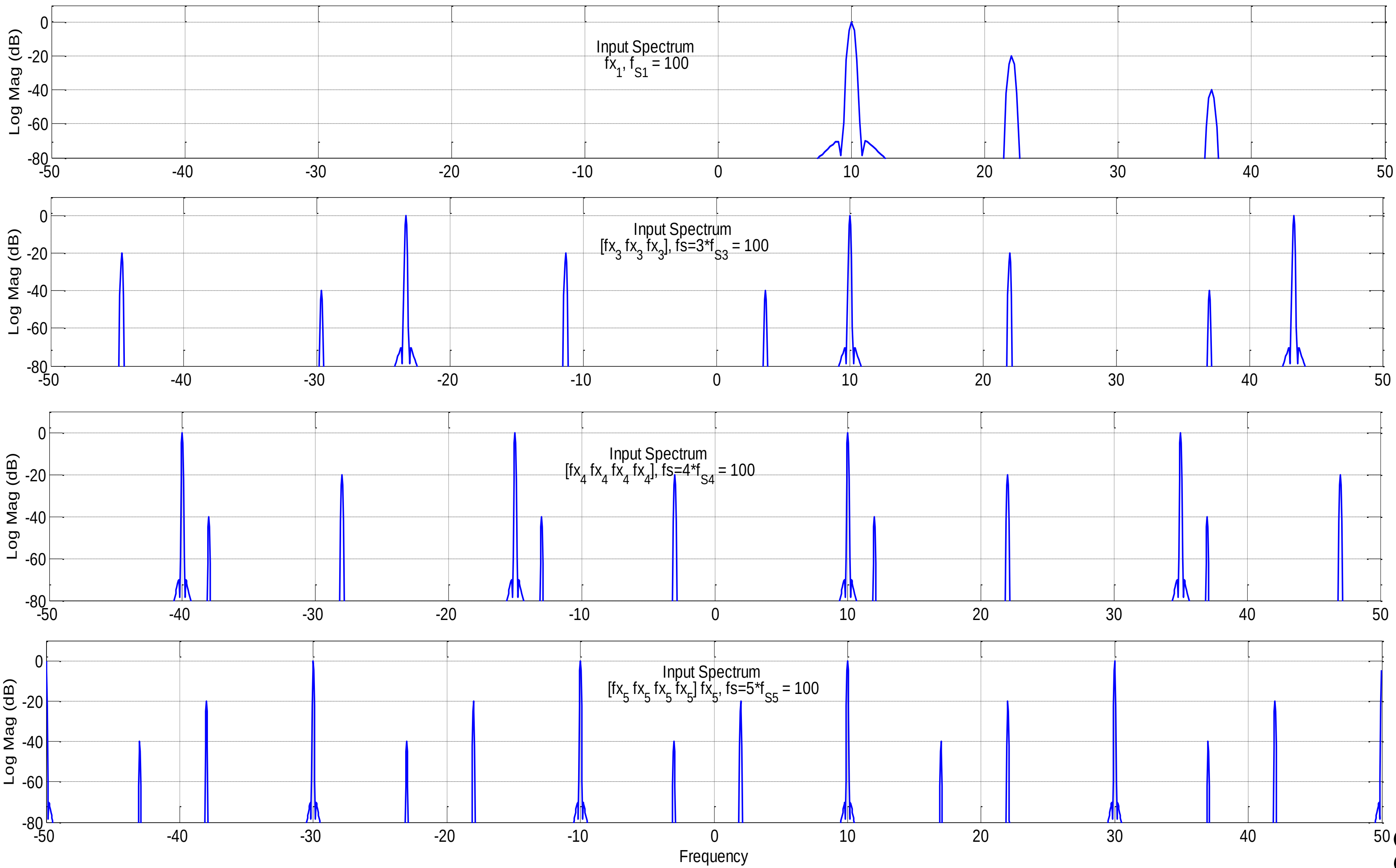
$$n = [n_1 N_2 M_2 + n_2 N_1 M_1] \text{ Mod}(N)$$
$$n = [n_1(3)(2) + n_2(5)(-1)] \text{ Mod}(20)$$

$$k = [k_1 N_2 + k_2 N_1] \text{ Mod}(N)$$
$$k = [n_1(3) + n_2(5)] \text{ Mod}(20)$$

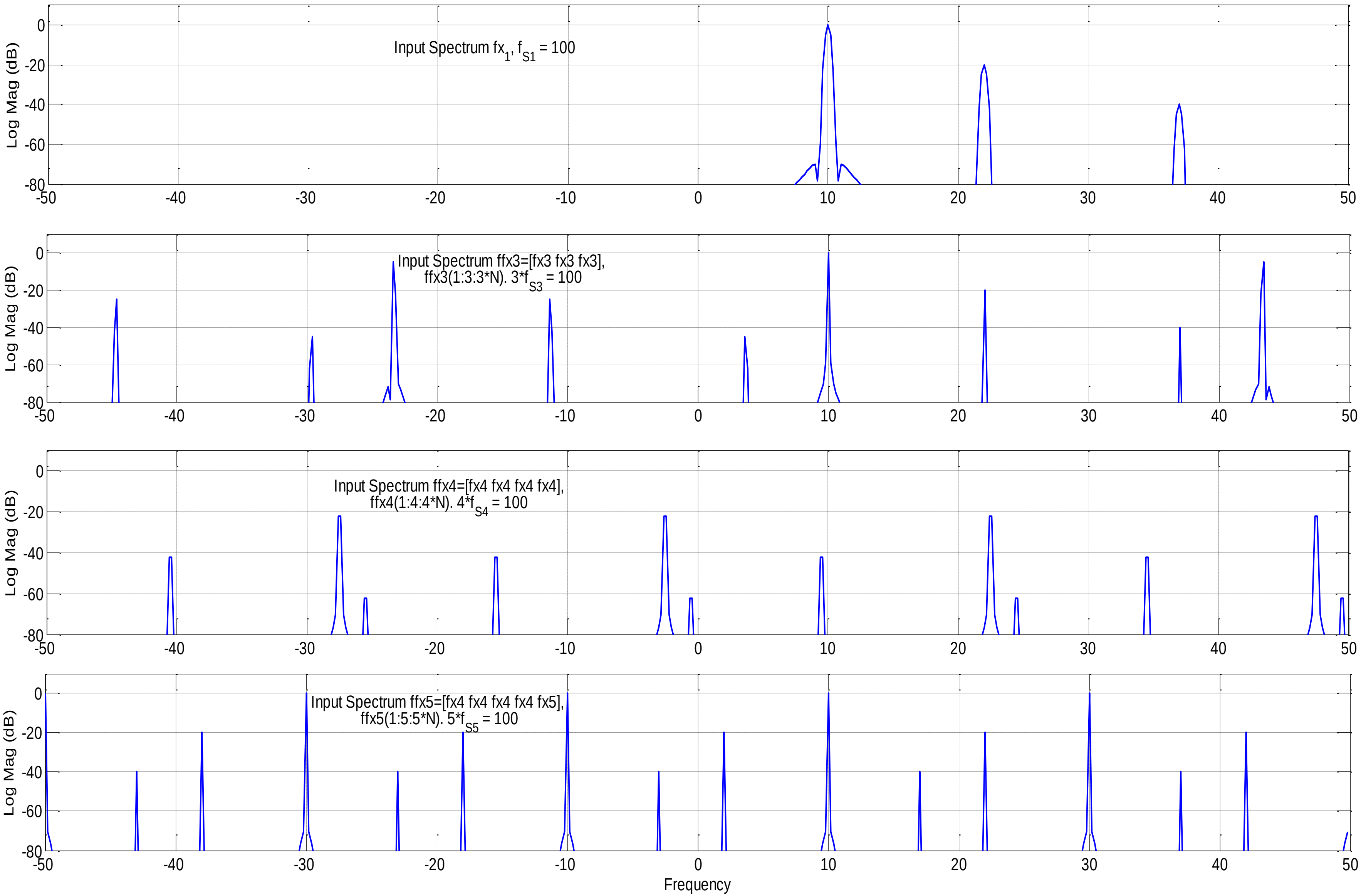
Sparse Spectrum Down Sampled to $f = f_S, f_S/3, f_S/4, f_S/5$

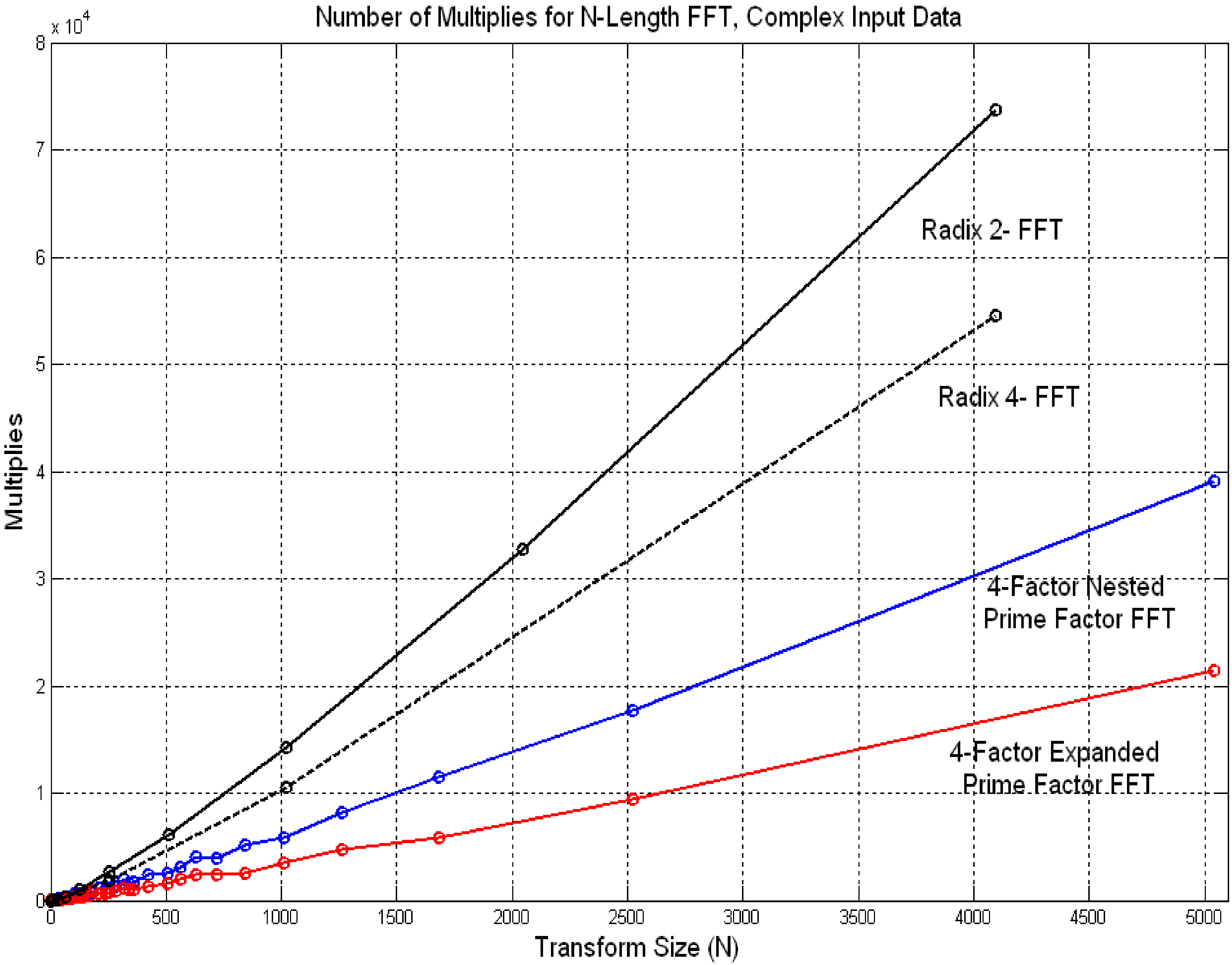


Sparse Spectrum Down Sampled to f = f_S, $f_S/3$, $f_S/4$, $f_S/5$ and Replicate Append by 3, 4, & 5 Respectively

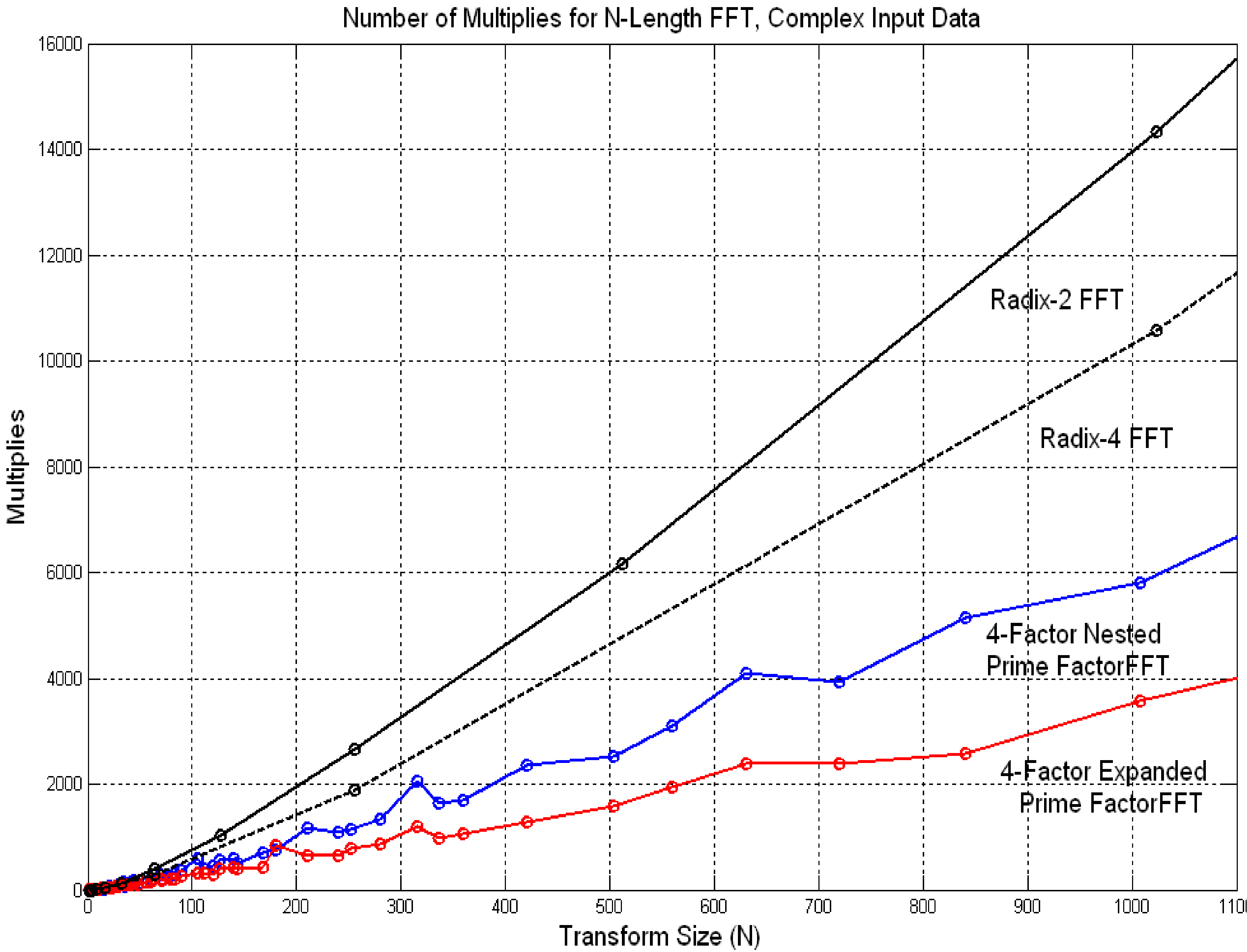


Sparse Spectrum Down Sampled to f = f_S, $f_S/3$, $f_S/4$, $f_S/5$ and Replicate Append by 3, 4, & 5 and Down Sampled by 3, 4, & 5 Respectively

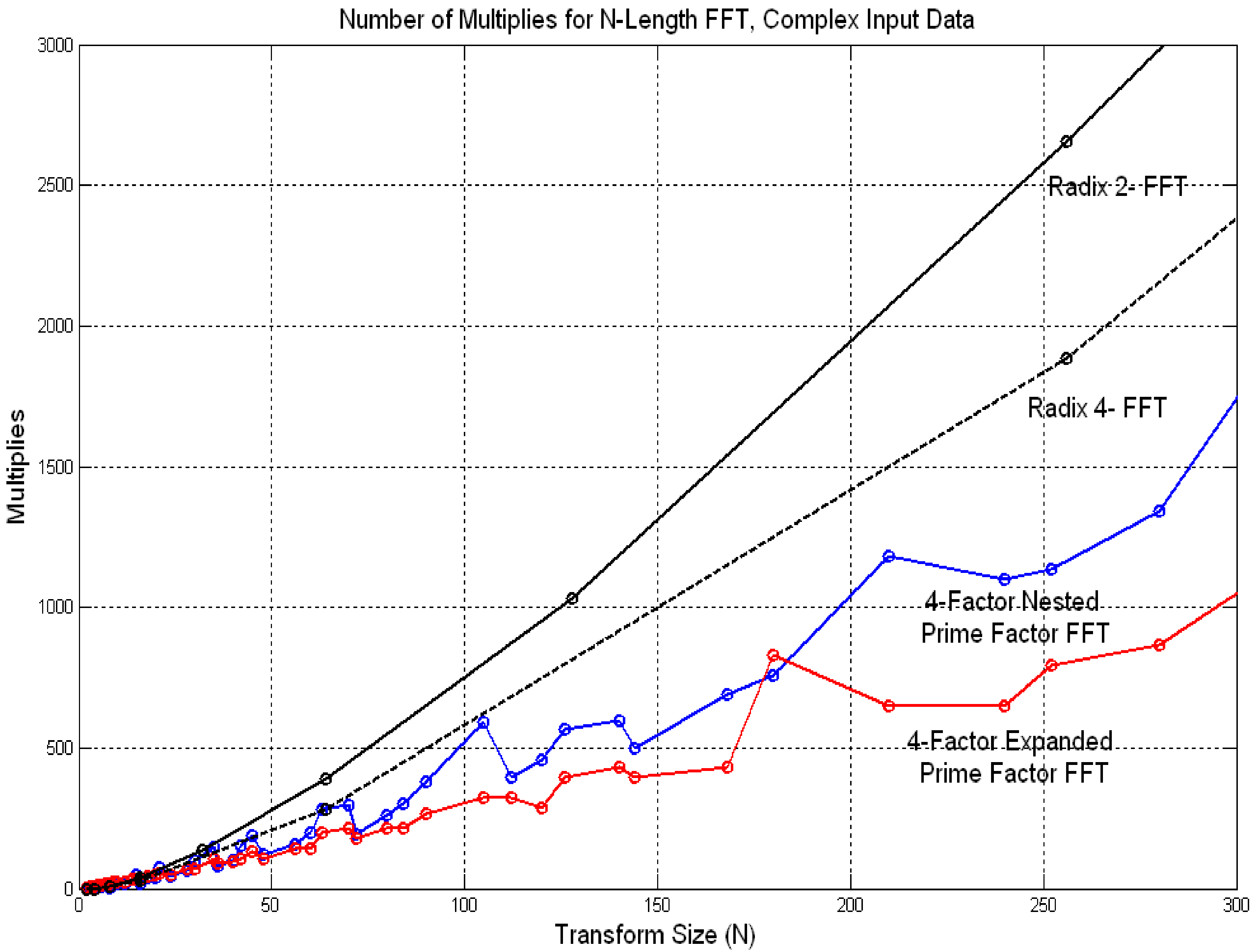




Number of Multiplies for N-Length FFT, Complex Input Data
x 10^4
8
7
6
5
4
3
2
1
0
Multiplies
0
500
1000
1500
2000
2500
3000
3500
4000
4500
5000
Transform Size (N)
Radix 2- FFT
Radix 4- FFT
4-Factor Nested
Prime Factor FFT
4-Factor Expanded
Prime Factor FFT



Number of Multiplies for N-Length FFT, Complex Input Data
Multiplies
Transform Size (N)
Radix-2 FFT
Radix-4 FFT
4-Factor Nested
Prime FactorFFT
4-Factor Expanded
Prime FactorFFT
0
2000
4000
6000
8000
10000
12000
14000
16000
100
200
300
400
500
600
700
800
900
1000
1100


Number of Multiplies for N-Length FFT, Complex Input Data
Multiplies
Transform Size (N)
0
500
1000
1500
2000
2500
3000
0
50
100
150
200
250
300
Radix 2- FFT
Radix 4- FFT
4-Factor Nested
Prime Factor FFT
4-Factor Expanded
Prime Factor FFT

BLUESTEIN FFT ALGORITHM

Convert DFT to Linear Convolver

$$H(k)=\sum_{n=0}^{N-1}h(n)\,e^{-j\frac{2\pi}{N}nk},\quad k=0,1,,....,N-1$$

$$(k-n)^2=k^2-2nk+n^2$$

$$-nk=\frac{1}{2}[(k-n)^2-k^2-n^2]$$

$$H(k)=\sum_{n=0}^{N-1}h(n)\,e^{-j\frac{2\pi}{2N}[(k-n)^2-k^2-n^2]}$$

$$=e^{+j\frac{2\pi}{2N}k^2}\sum_{n=0}^{N-1}h(n)\,e^{-j\frac{2\pi}{2N}[(k-n)^2-n^2]}$$

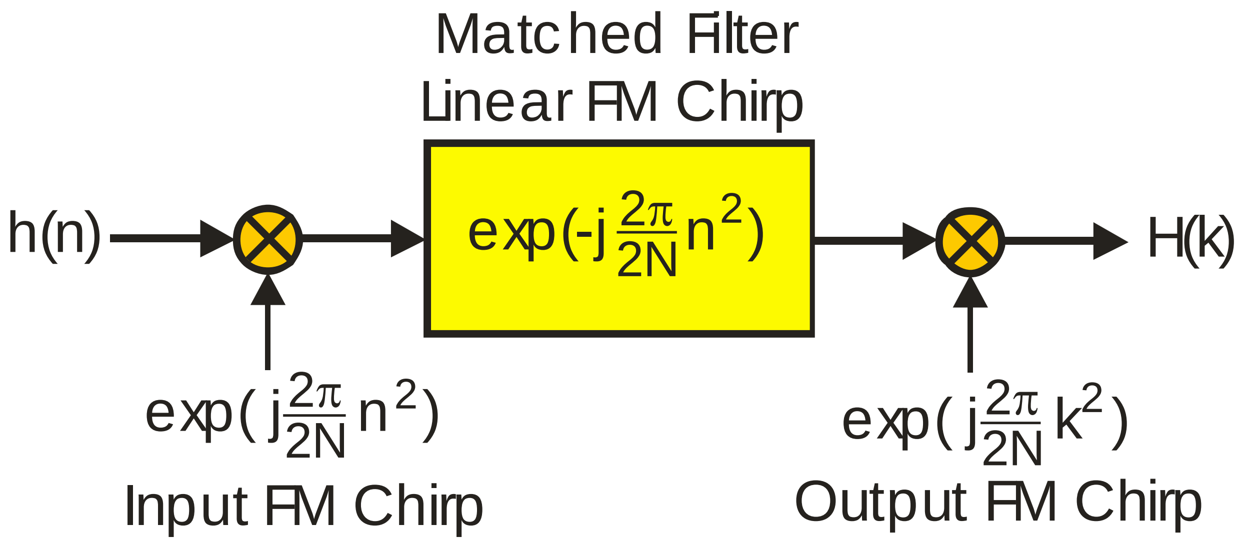
De-Chirp Spectrum

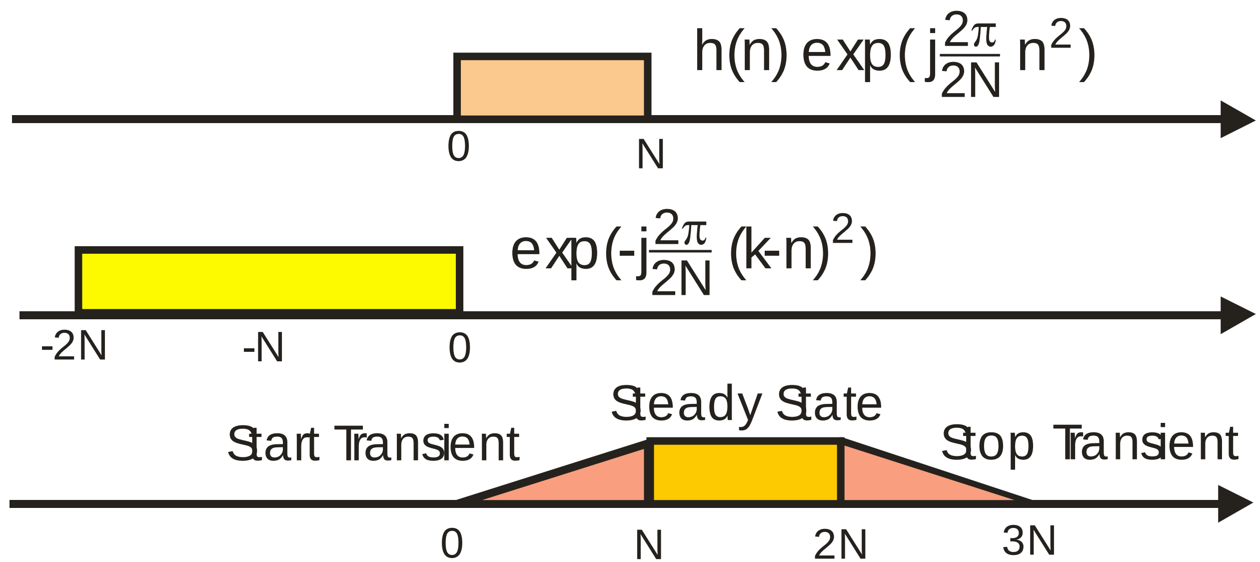
Chirp the Input Signal

$$=e^{+j\frac{2\pi}{2N}k^2}\sum_{n=0}^{N-1}h(n)\,e^{+j\frac{2\pi}{2N}n^2}\,e^{-j\frac{2\pi}{2N}(k-n)^2}$$

Chirp Matched Filter

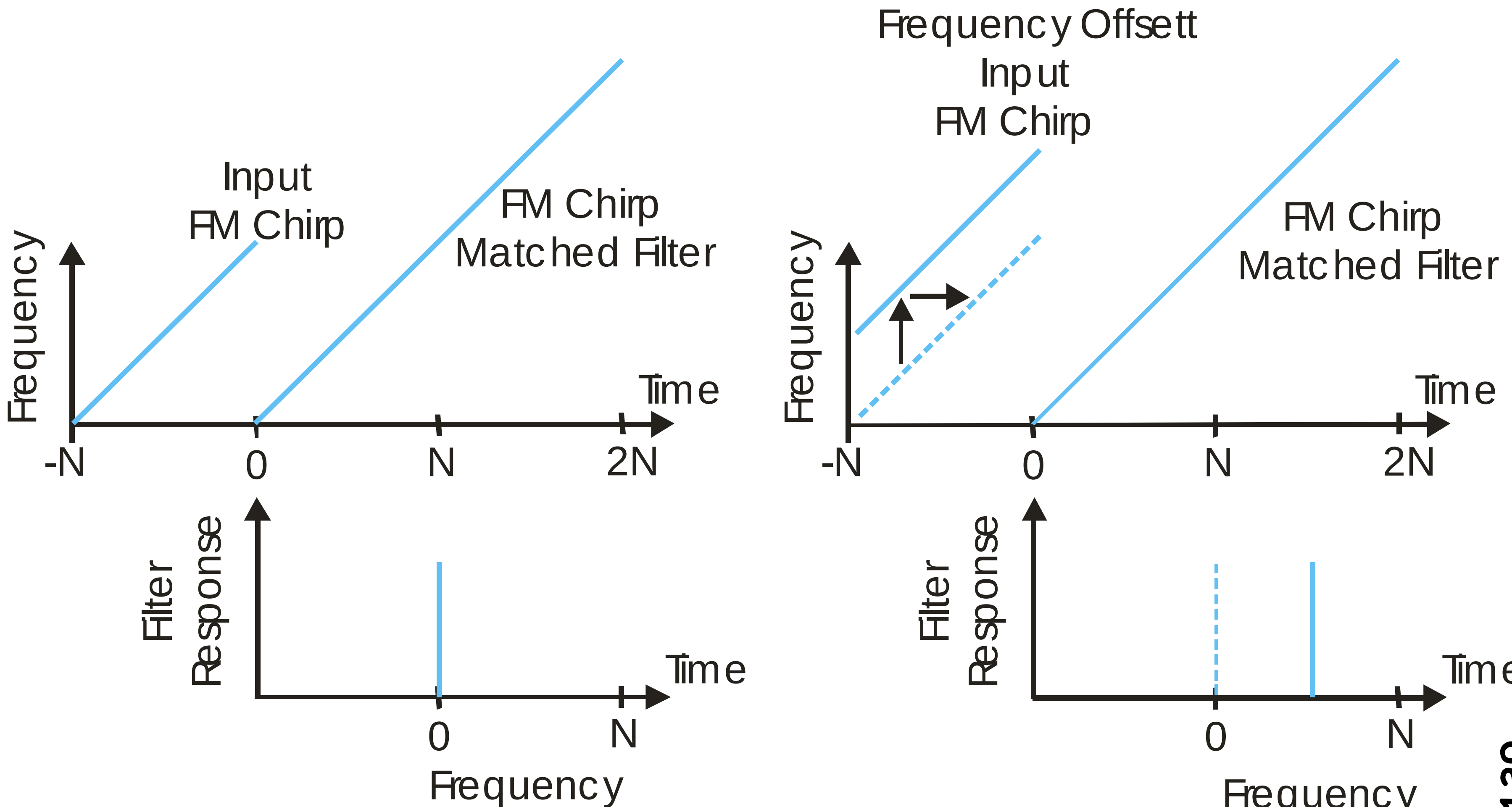
BLUESTEIN CHIRP-TRANSFORM





FREQUENCY PROPORTIONAL TO TIME:

SUCCESSIVE OUTPUTS FROM FILTER ARE DFT FREQUENCY BINS



Interesting Observation When N is Prime

FIVE POINT TRANSFORM

$$\begin{bmatrix} H_0 \\ H_1 \\ H_2 \\ H_3 \\ H_4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & W^1 & W^2 & W^3 & W^4 \\ 1 & W^2 & W^4 & W^1 & W^3 \\ 1 & W^3 & W^1 & W^4 & W^2 \\ 1 & W^4 & W^3 & W^2 & W^1 \end{bmatrix} \begin{bmatrix} h_0 \\ h_1 \\ h_2 \\ h_3 \\ h_4 \end{bmatrix}$$

Non Trivial
(Non Zero Index)
Sub-Matrix

$$W = \exp(-j\frac{2\pi}{5})$$

CONVERT TRANSFORM MATRIX TO CIRCULANT MATRIX

$$\begin{bmatrix} W^1 & W^2 & W^3 & W^4 \\ W^2 & W^4 & W^1 & W^3 \\ W^3 & W^1 & W^4 & W^2 \\ W^4 & W^3 & W^2 & W^1 \end{bmatrix}$$

4 by 4 Non-Zero Index
Sub-Matrix Of
5-point FFT Matrix

$$\begin{bmatrix} W^1 & W^2 & W^3 & W^4 \\ W^2 & W^4 & W^1 & W^3 \\ W^4 & W^3 & W^2 & W^1 \\ W^3 & W^1 & W^4 & W^2 \end{bmatrix}$$

Interchange
Rows 3 and 4

2 is Primitive Modulo 5

2^{k1}<=>k	2^{-n1}<=>n
$2^0=1$	$2^{-0}=1$
$2^1=2$	$2^{-1}=3$
$2^2=4$	$2^{-2}=4$
$2^3=3$	$2^{-3}=2$

$$\begin{bmatrix} W^1 & W^3 & W^4 & W^2 \\ W^2 & W^1 & W^3 & W^4 \\ W^4 & W^2 & W^1 & W^3 \\ W^3 & W^4 & W^2 & W^1 \end{bmatrix}$$

Reorder Columns
1, 2, 3, 4 to 1, 3, 4, 2

Each Row is Now a
Circular Shift of
the Previous Row

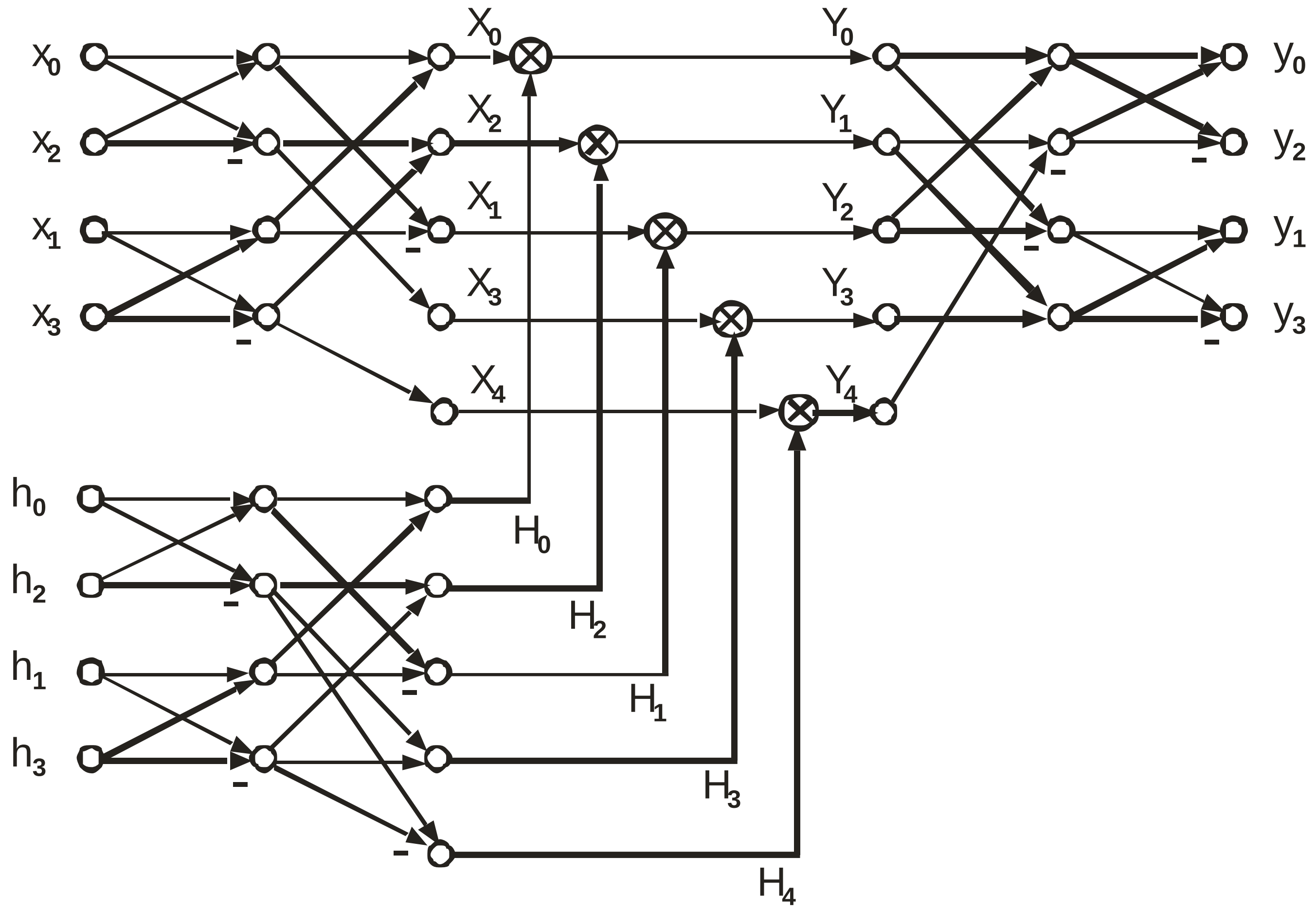
FOUR POINT WINOGRAD CIRCULAR CONVOLVER

$[Y_0\ Y_1\ Y_2\ Y_3] = [X_0\ X_1\ X_2\ X_3] \circledast [H_0\ H_1\ H_2\ H_3]$

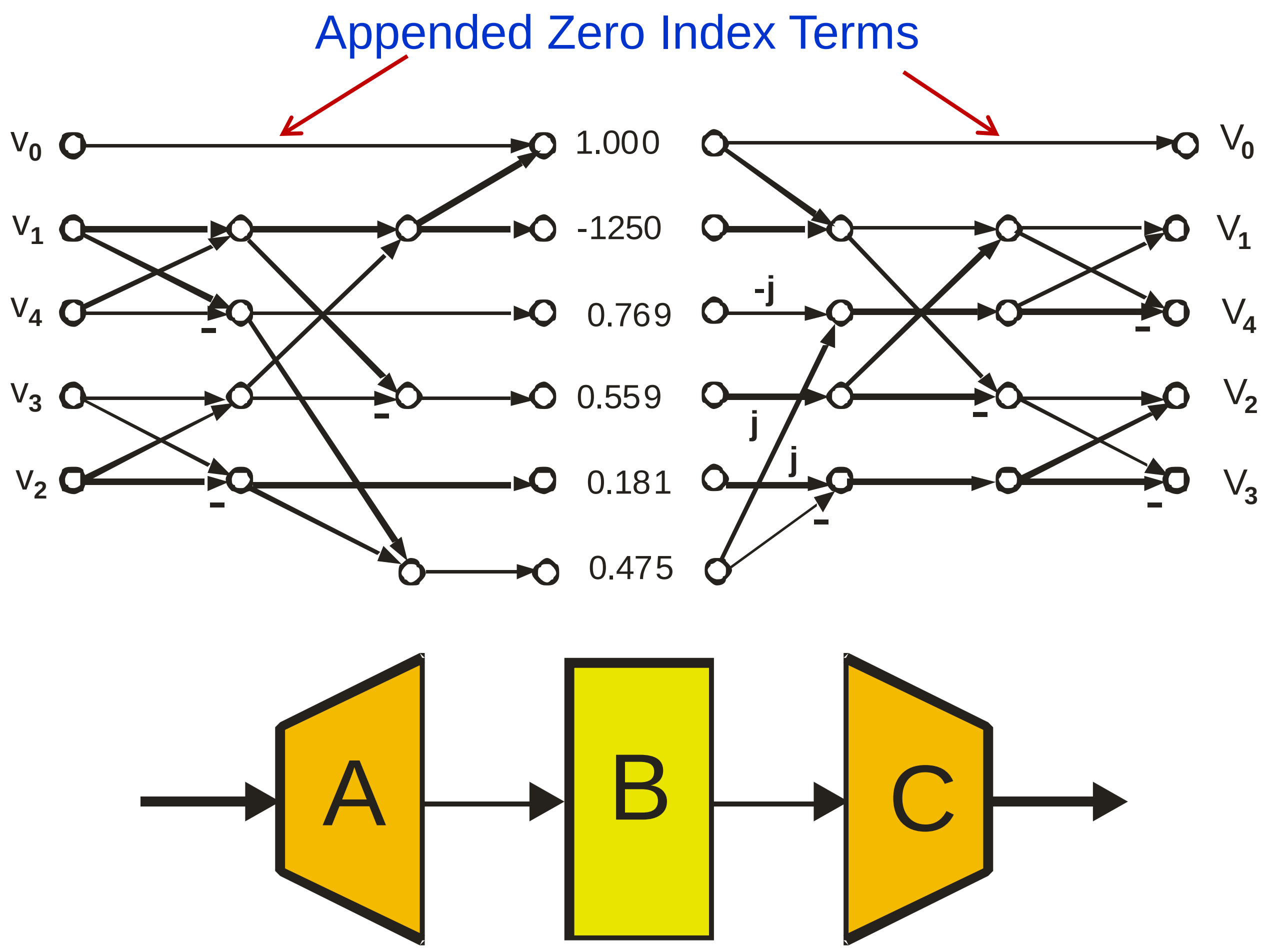
$$
\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 0 & -1 \\ 1 & -1 & 1 & -1 & 0 \\ 1 & 1 & -1 & 0 & 1 \\ 1 & -1 & -1 & 1 & 0 \end{bmatrix} \begin{bmatrix} H_0 & & & & \\ & H_1 & & & \\ & & H_2 & & \\ & & & H_3 & \\ & & & & H_4 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix}
$$

$$
\begin{bmatrix} H_0 \\ H_1 \\ H_2 \\ H_3 \\ H_4 \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 2 & 0 & -2 & 0 \\ 2 & -2 & -2 & 2 \\ 2 & 2 & -2 & -2 \end{bmatrix} \begin{bmatrix} h_0 \\ h_1 \\ h_2 \\ h_3 \end{bmatrix}
$$

SIGNAL FLOW GRAPH OF FOUR POINT WINOGRAD CIRCULAR CONVOLVER



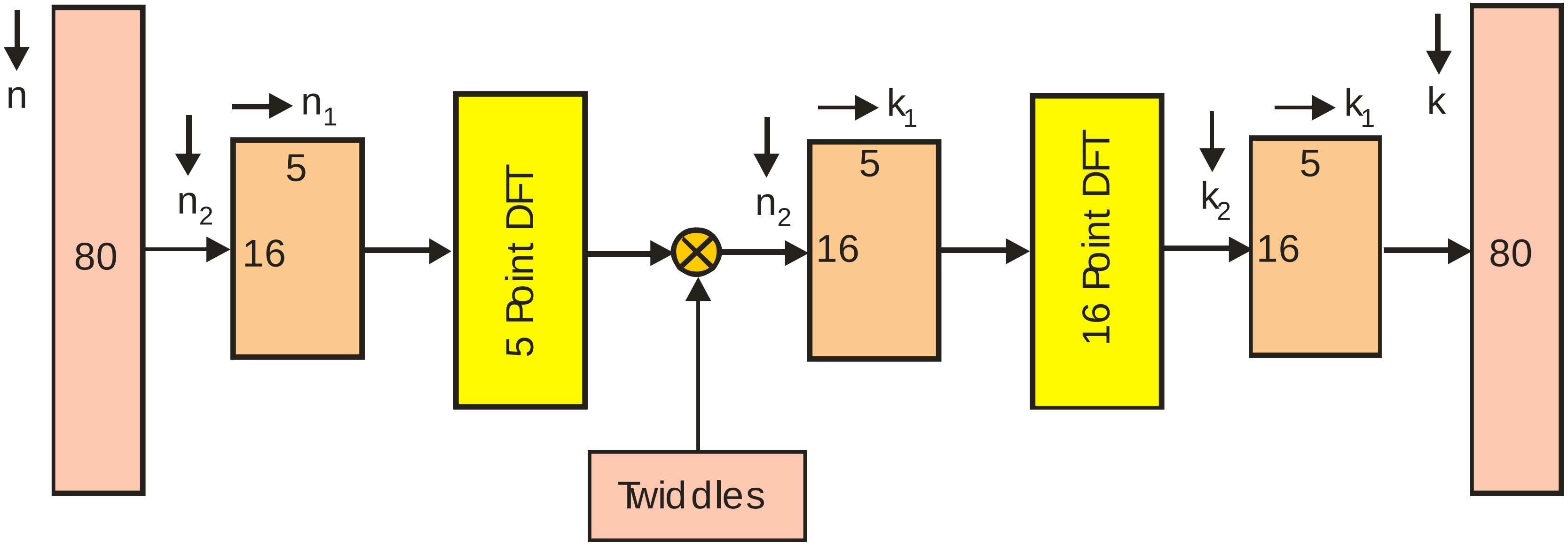
Five Point Winograd FFT



Some Elementary Winograd Fourier Transforms

N	M(N)	MULT	ADD
2	2	0	2
3	3	2	6
4	4	0	8
5	6	5	17
7	9	8	38
8	8	2	26
9	13	10	44
11	21	20	84
13	21	20	94
16	18	10	74
17	36	35	157
19	39	38	186

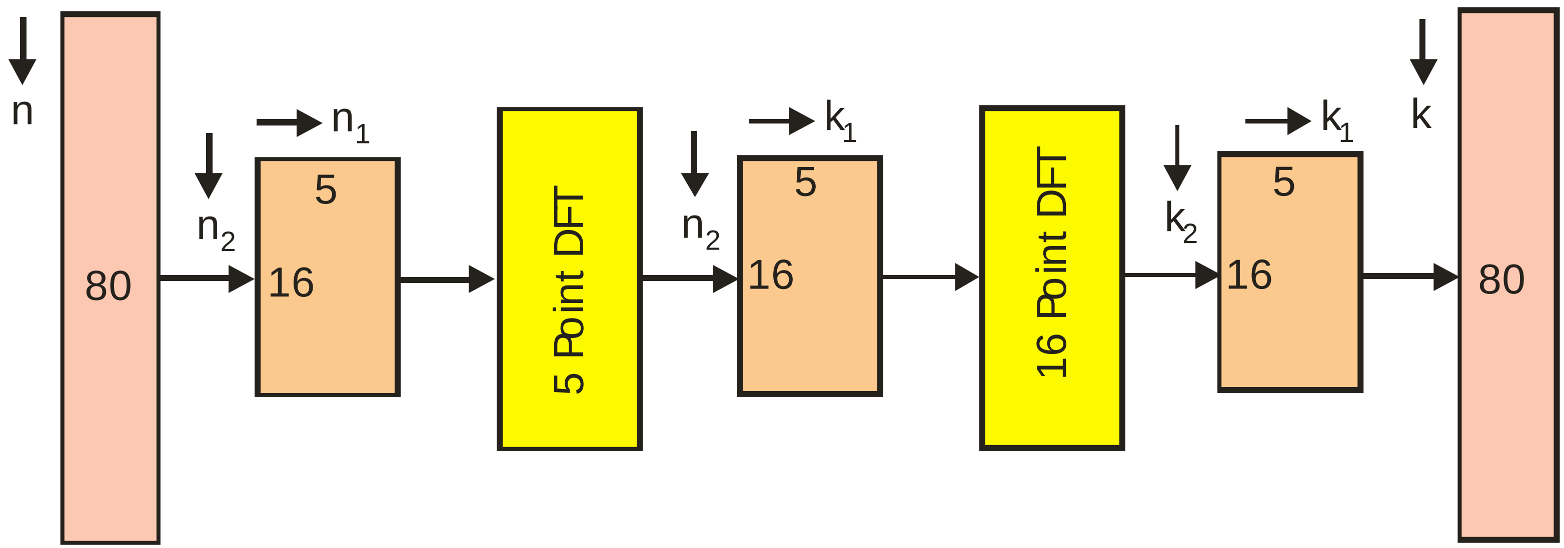
80 POINT COOLEY-TUKEY FFT



5 Point DFT	-	16 Times:	16 x 25
5 Twiddles	-	16 Times:	16 x 5
16 Point DFT	-	5 Times:	5 x 256

1760 Complex Multiplies
1680 Complex Additions

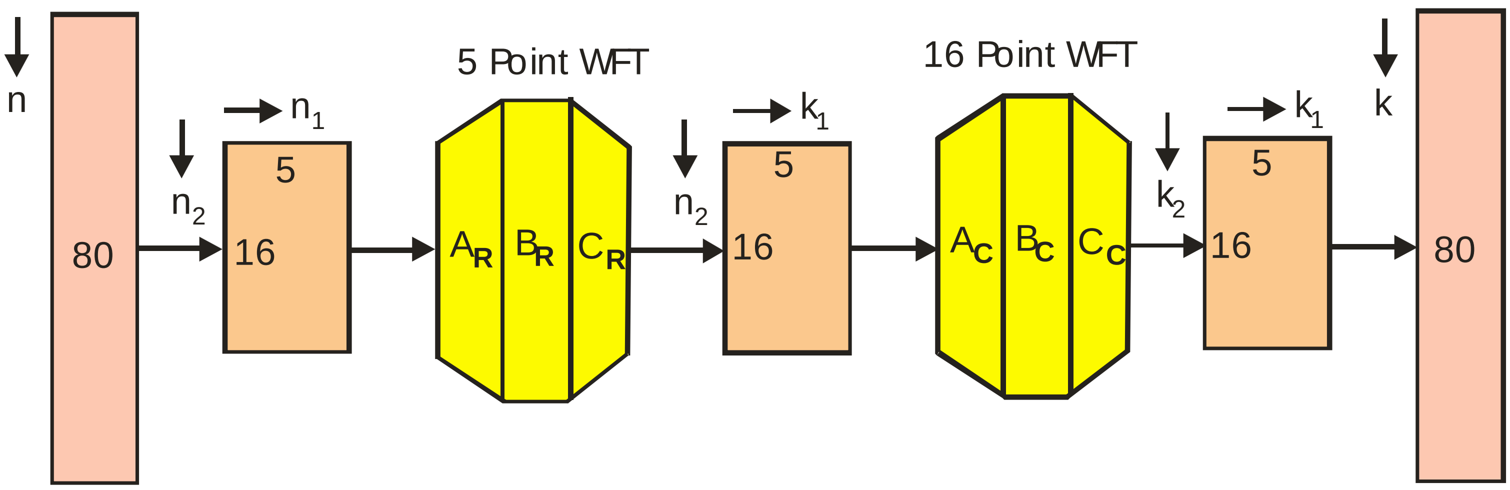
80 POINT GOOD-THOMAS FFT



5 Point DFT - 16 Times: 16 x 25
16 Point DFT - 5 Times: 5 x 256

1680 Complex Multiplies
1680 Complex Additions

80 POINT G-T WITH WINOGRAD FFT



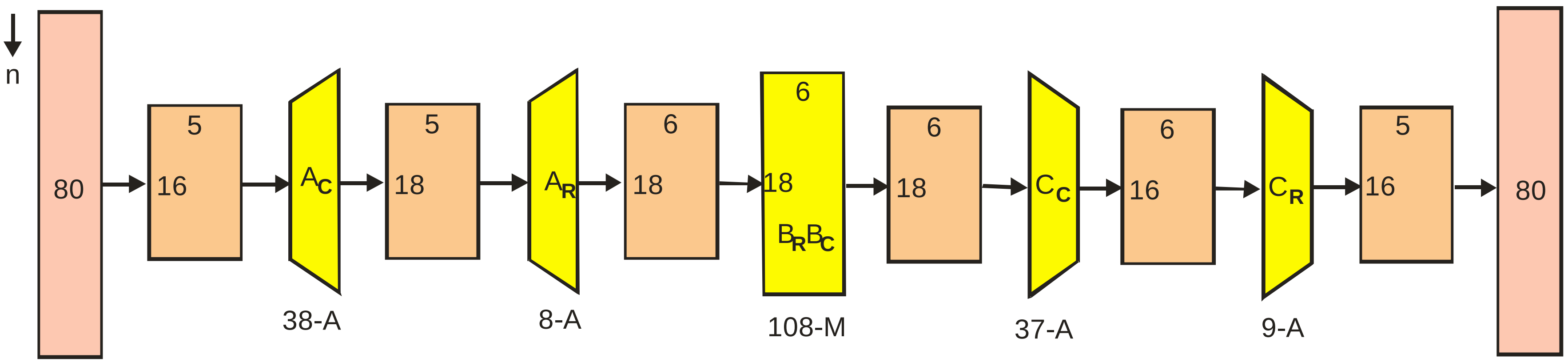
5 Point WFT - 16 Times: 16 x (5-M, 17-A)

16 Point WFT - 5 Times: 5 x (10-M, 74-A)

} x 2

260 Real Multiplies
1280 Real Additions

80 POINT NESTED WINOGRAD FFT



$$[5\,A_C + 18\,A_R + 1B_R B_C + 6\,C_C + 16\,C_R] \times 2$$

$$2[\,5(38\text{-}A) + 18(8\text{-}A) + 1(108\text{-}M) + 6(37\text{-}A) + 16(9\text{-}A)]$$

$$= 2[700\text{-}A + 108\text{-}M] = \begin{cases} 216 \text{ Real Multiplies} \\ 1400 \text{ Real Additions} \end{cases}$$

COMPARISON OF FFT ALGORITHMS (COMPLEX INPUT DATA)

Good-Thomas, Winograd FFT | Radix-2 Cooley-Tukey FFT

Block length	Factors	Real Mult	Real Adds	Block Length	Real Mult	Real Adds
30	2x3x5	72	384	32	320	480
48	3x16	92	636			
60	3x4x5	144	888	64	768	1,152
91	7x13	318	2,648			
120	3x5x8	288	2,076	128	1,792	2,688
168	3x7x8	432	3,492			
240	3x5x16	648	5,012	256	4,096	6,144
420	3x4x5x7	1,296	11,352			
504	7x8x9	1,584	14,642	512	9,216	6,144
840	3x5x7x8	2,592	24,804			
1,008	7x9x16	3,564	34,920	1,024	20,480	30,720
2,520	5x7x8x9	9,504	100,188	2,048	45,056	67,584
10,920	3 x5x7x8x13	38,760	320,196	8,192	212,992	319,488

REFERENCES

- I.J. Good (1958)
  The interaction algorithm and practical Fourier analysis".
  Journal of the Royal Statistical Society, Series B **20** (2): 361–372.

- L.H. Thomas (1963)
  Using a Computer to Solve problems in Physics,
  Ginn & Co., Boston, Mass, 1963

- Cooley, James W., and John W. Tukey (1965)
  An algorithm for the machine calculation of complex Fourier series
  Math. Of Computer. Vol. 19, April 1965, 297-301

- C. Rader(1965),
  Discrete Fourier Transform when the Number of Data Samples is Prime
  Proceedings of the IEEE Vol.5, No. 6, June 1968, 1107-1108

- Leo Bluestein (1970)
  A Linear Filtering Approach to the Computation of Discrete Fourier Transform
  IEEE Trans Audio & Electroacoustics, Vol. AU-18, No. 4, 451-455

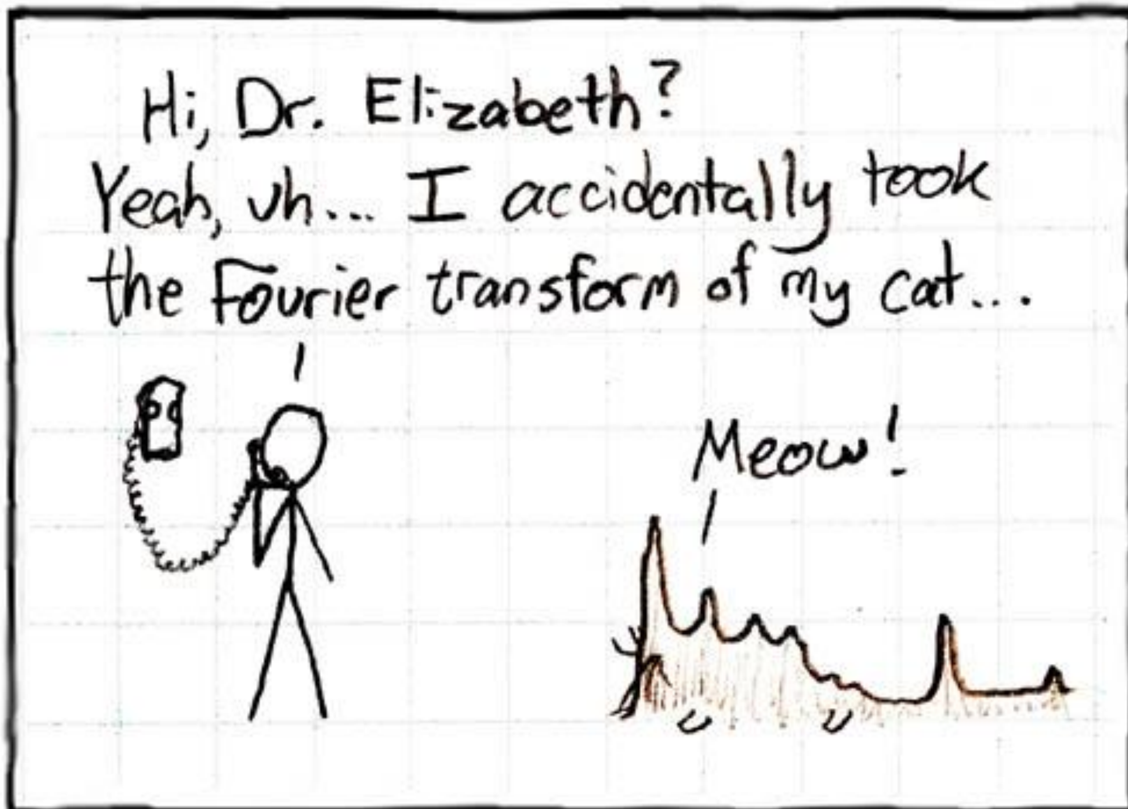
- S. Winograd (1978)
  On computing the discrete Fourier transform
  Math. Comp., 32, pp. 175–199

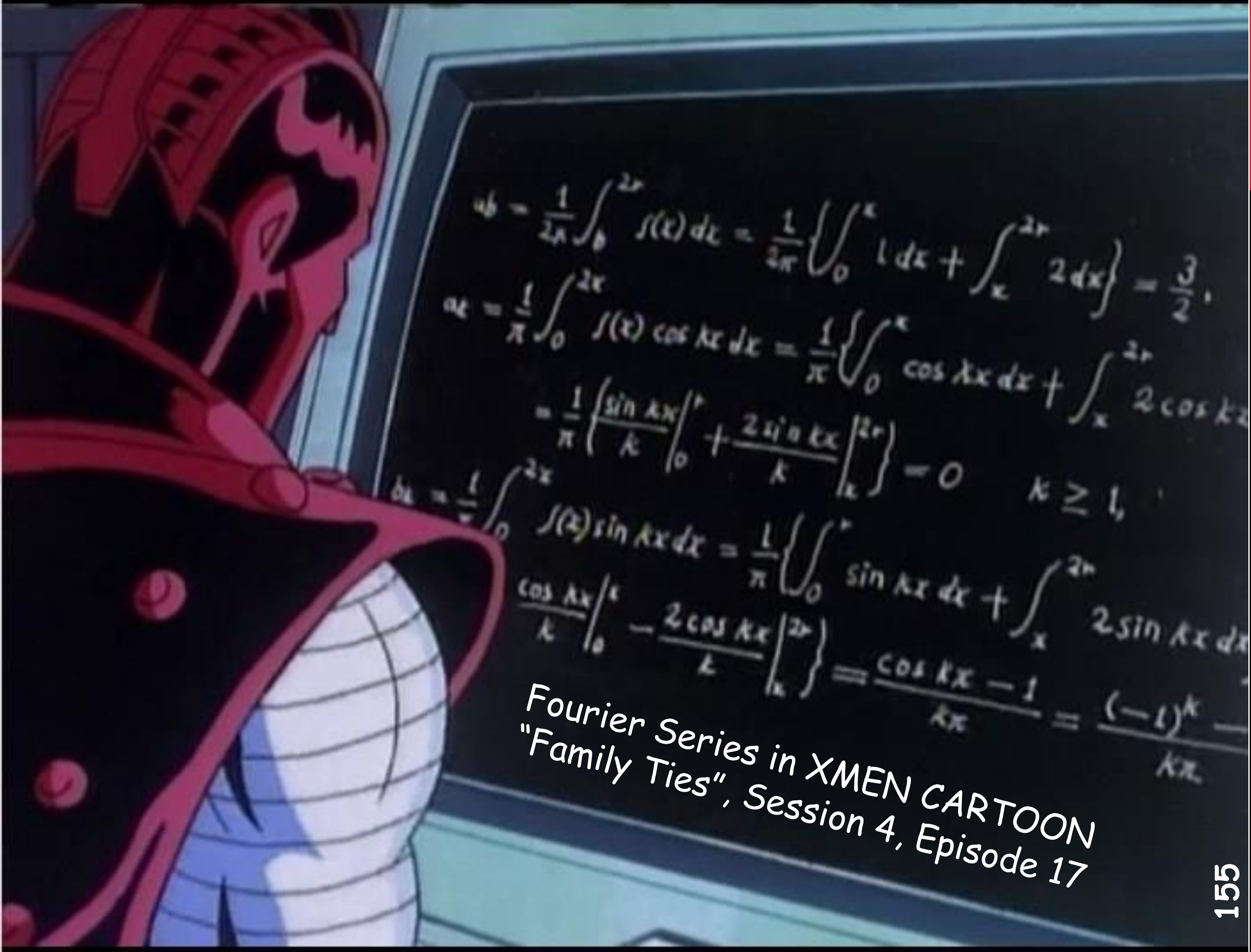
TEXTBOOK REFERENCES

- E. Brigham, *Fast Fourier transform and Its Applications*
- R. Blahut, *Fast Algorithms for Signal Processing*
- H. Nussbaumer, *Fast Fourier Transform and Convolution Algorithms*
- C. Burrus & T. Parks, *DFT/FFT and Convolution Algorithms and Implementation*
- D. Elliot & K. Rao, *Fast Fourier Transforms, Analysis, Applications*
- R Ramirez, *The FFT: Fundamentals and Concepts*

- R. Bracewell, *The Fourier Transform & Its Applications*

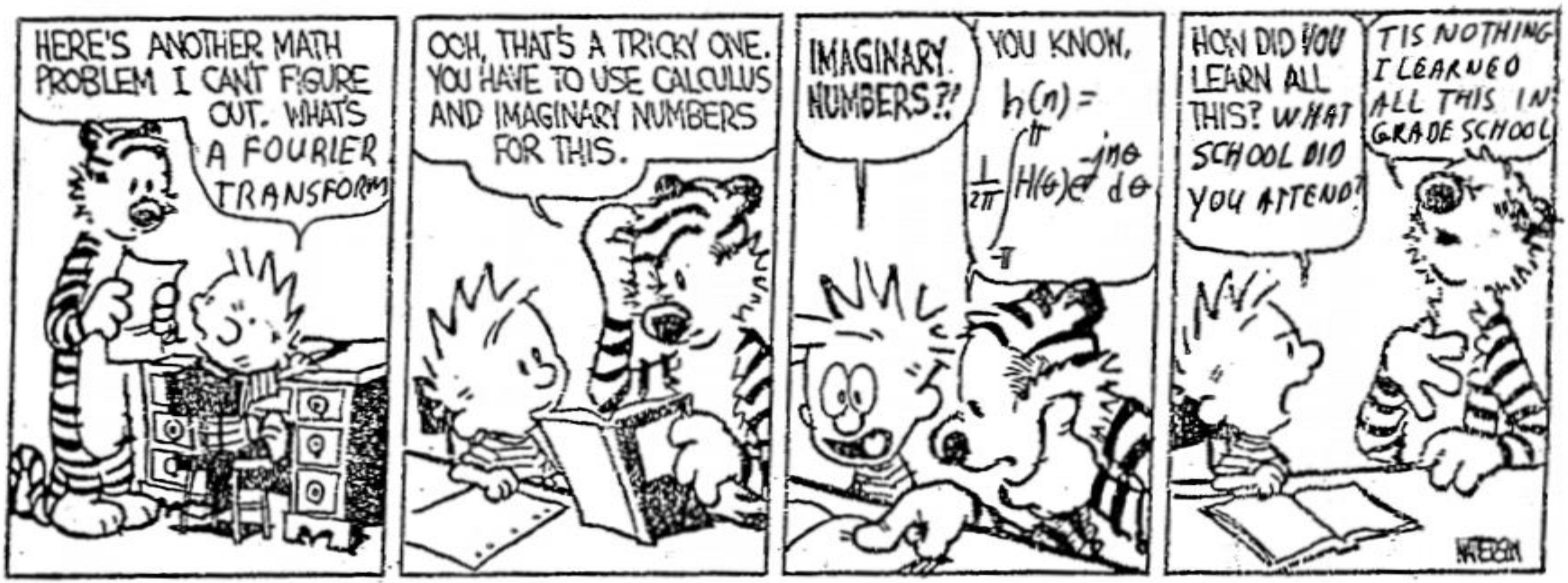
Closing Cartoons

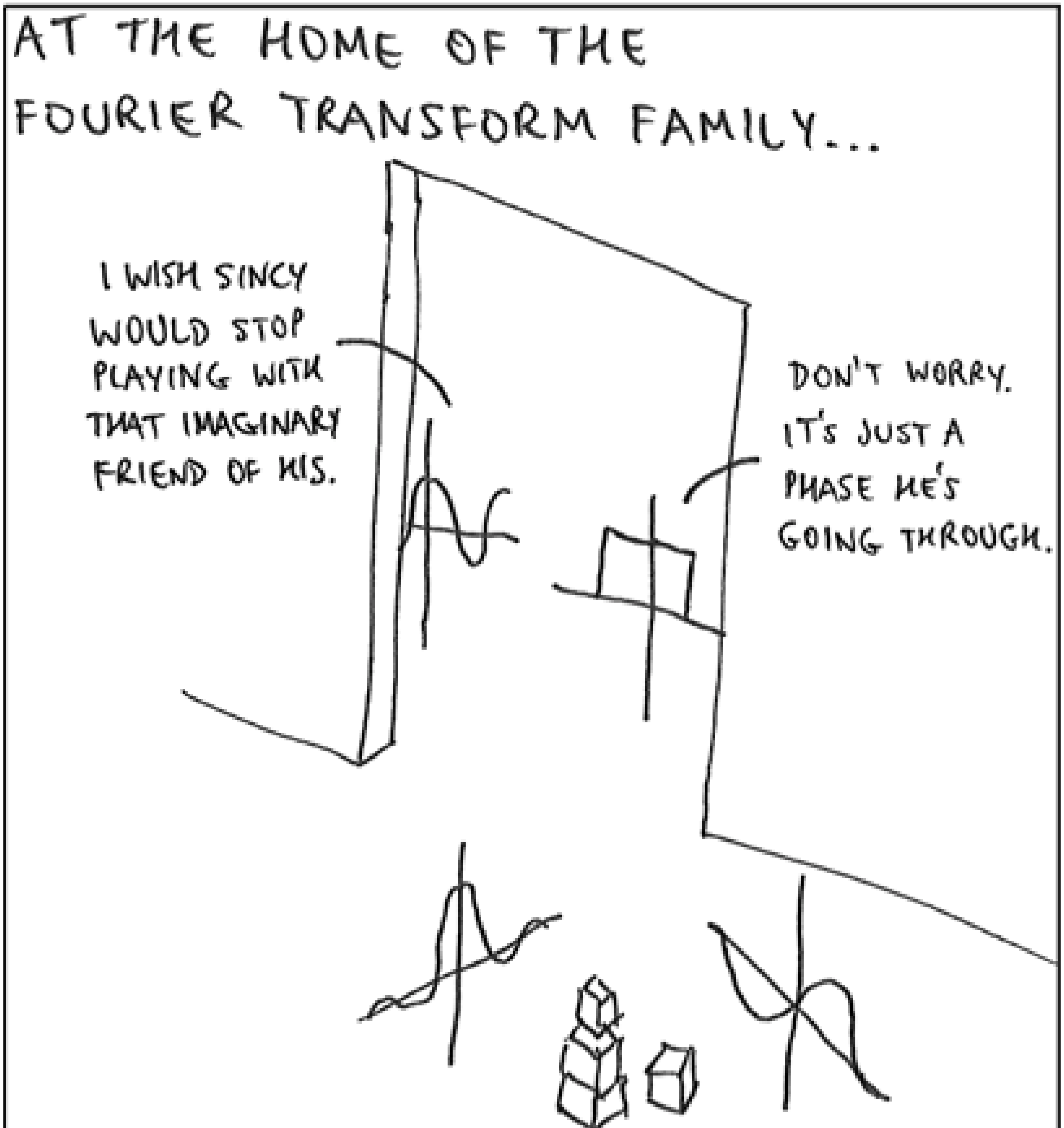

Hi, Dr. Elizabeth?
Yeah, uh... I accidentally took the Fourier transform of my cat...
Meow!



Fourier Series in XMEN CARTOON
"Family Ties", Session 4, Episode 17

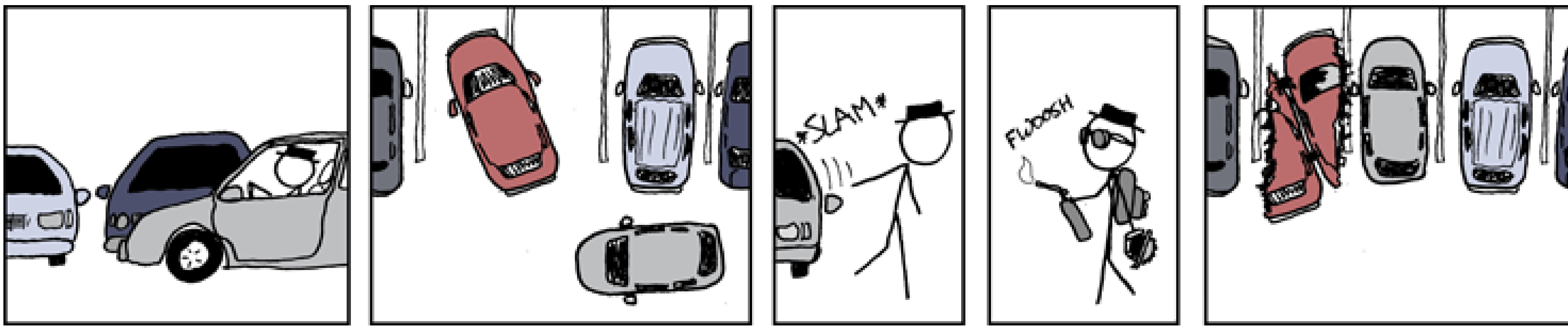
What School Did You Attend?
(Calvin and Hobbs)



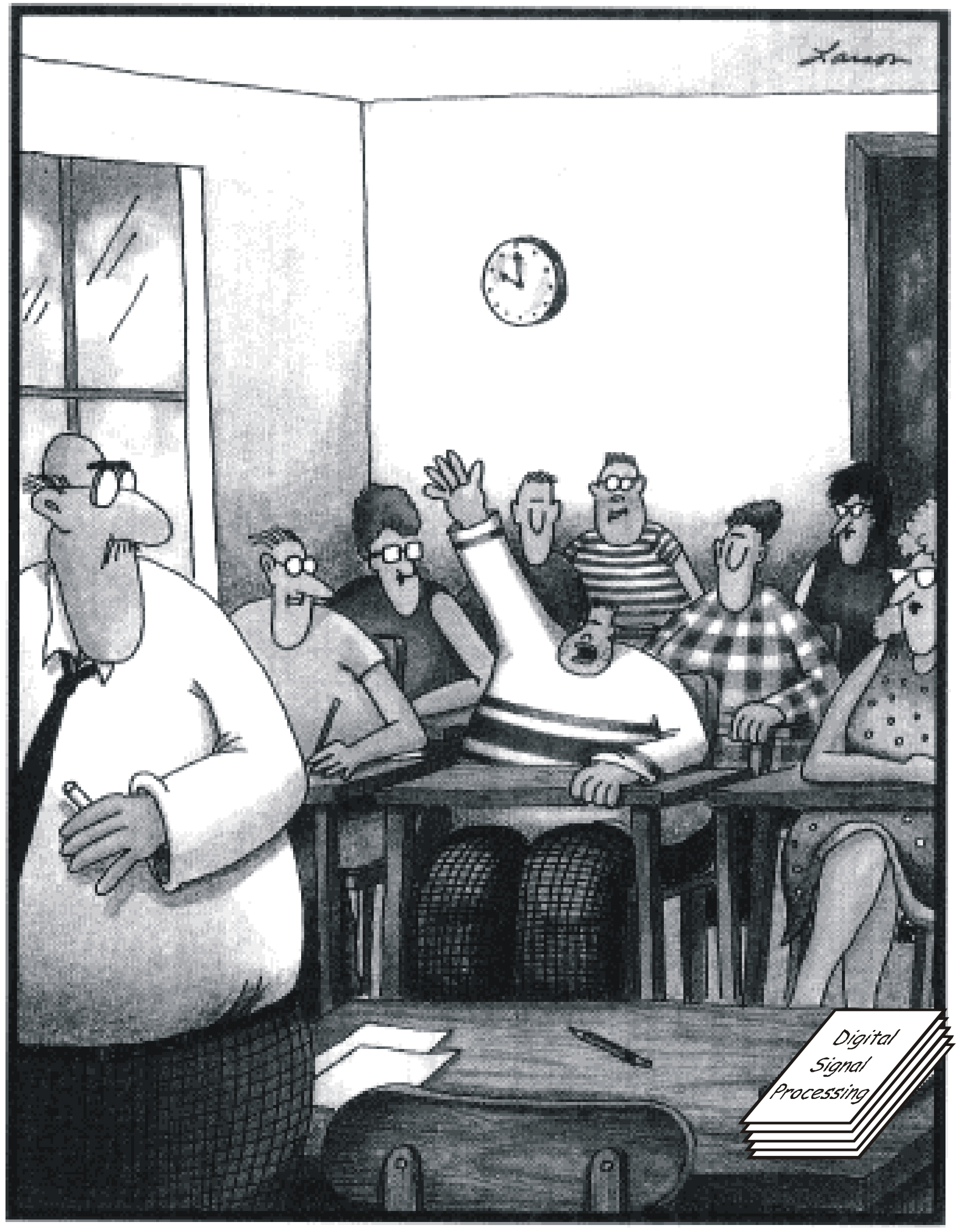

AT THE HOME OF THE FOURIER TRANSFORM FAMILY...
I WISH SINCY WOULD STOP PLAYING WITH THAT IMAGINARY FRIEND OF HIS.
DON'T WORRY. IT'S JUST A PHASE HE'S GOING THROUGH.

A CARTOON THAT HAS NOTHING TO DO WITH THE FFT.

TO SEE THE HUMOR HERE
YOU MAY HAVE HAD TO
SEARCH FOR A PARKING SPACE AT SDSU



To see the humor here
you may have had to
search for a parking space at SDSU

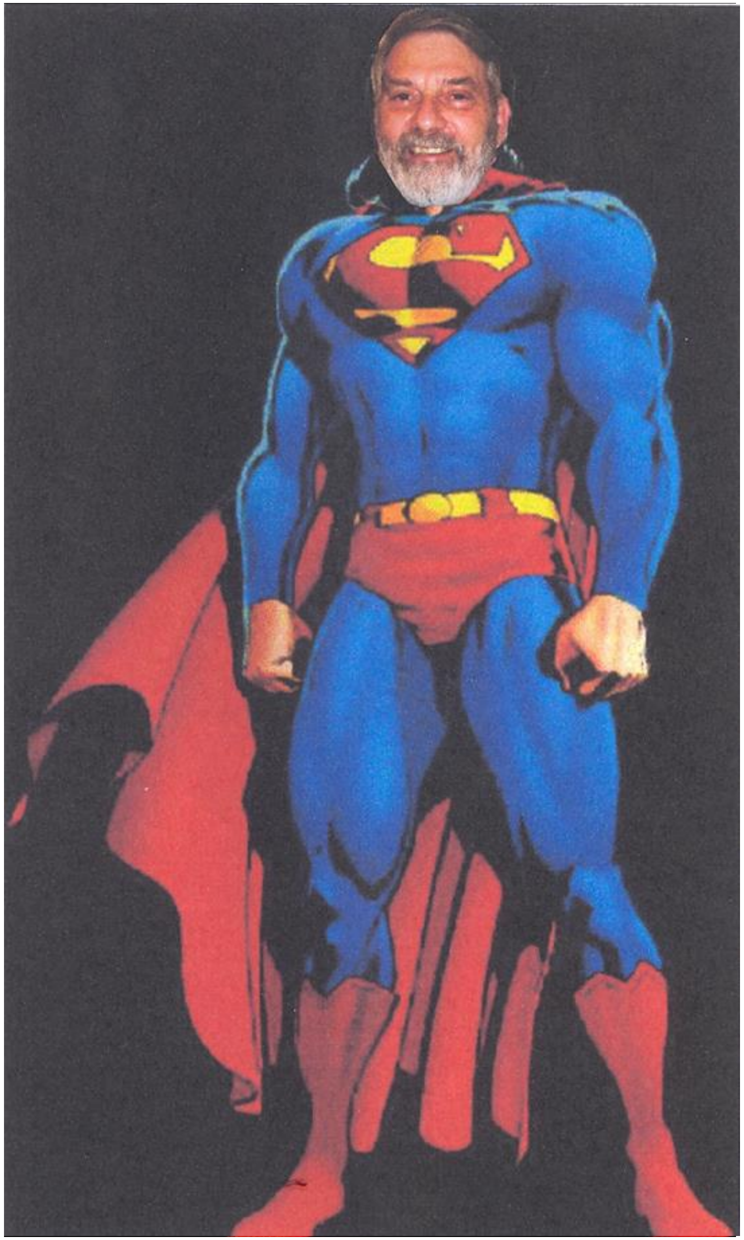


Professor harris, may I be excused?
My brain is full.

We are now open For Questions

THANKS FOR BEING A GOOD AUDIENCE!

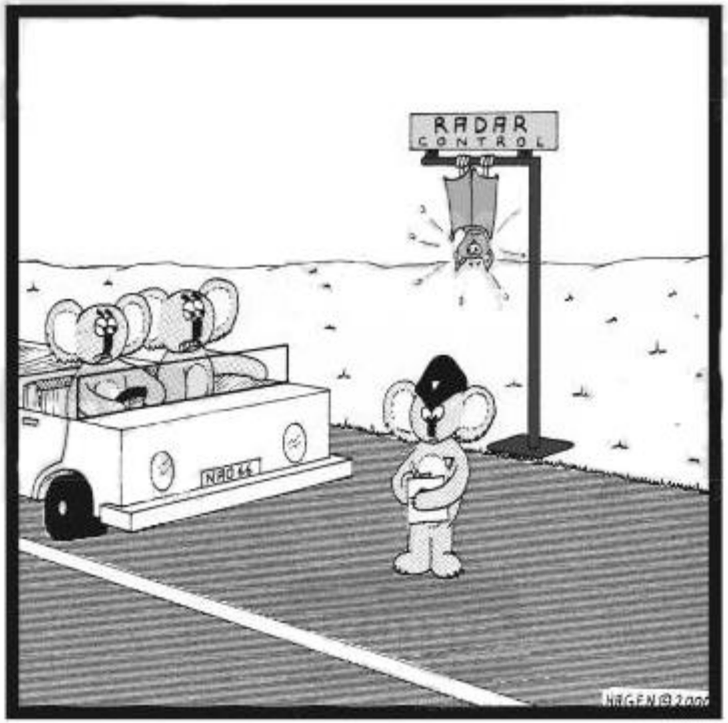
One More Slide



SOFTWARE DEFINED RADIO MAN

Is Open For Questions


INFORMATION
SUPER HIGHWAY
SPEED LIMIT
5,000,000,000
BYTES PER SECOND
ENFORCED BY RADAR


RADAR
CONTROL