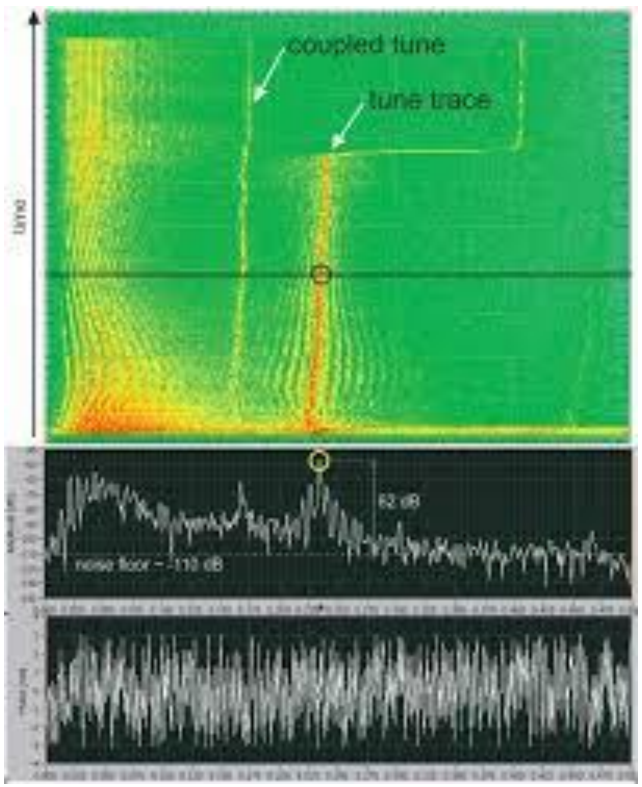
Basic Introductory Course, Video Course 2021

Time & Frequency Domain Measurements

H.Schmickler, CERN



Using several slides from:

M.Gasior (CERN)
R.Jones (CERN)
T.Lefevre (CERN)
H. Damerau (CERN)
S.Zorzetti (FNAL)

The CERN Accelerator School

Copyright statement and speaker's release for video publishing

Outline

In red: items dropped from 2 hour version

- Introduction: What Is time domain and frequency domain?
- Fourier synthesis and Fourier transform
- Time domain sampling of electrical signals (→ ADCs)
- Bunch signals in time and frequency domain
  - a) single bunch single pass
  - b) single bunch multi pass (circular accelerator)
  - c) multi bunch multi pass (circular accelerator) → not this time
  - d) Oscillations within the bunch (head-tail oscillations) → not this time
- Fourier transform of time sampled signals
  - basics, aliasing, windowing
- Methods to improve the frequency resolution
  - a) interpolation
  - b) fitting (the NAFF algorithm)
  - c) influence of signal to noise ratio
  - d) special case: no spectral leakage + IQ sampling
- Analysis of non stationary spectra:
  - STFT (:= Short time Fourrier transform) (Gabor transform) also called: Sliding FFT, Spectogram
  - wavelet analysis
  - PLL tune tracking

Complete 2hour version of course

Slides:

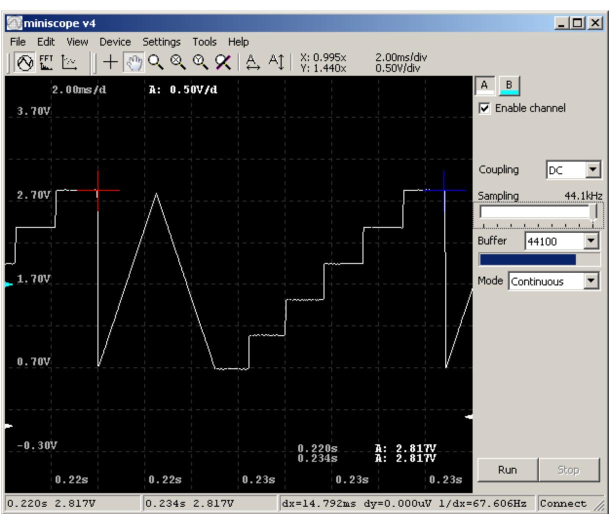
https://indico.cern.ch/event/808940/contributions/3553569/attachments/1906422/3149268/timefrequency12.pptx

Writeup:

arXiv:2009.14544v1 [physics.acc-ph]

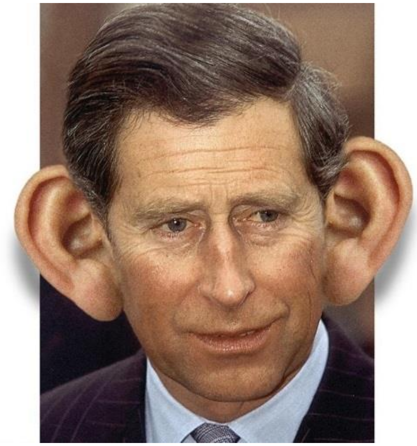
Introduction 1/3

- At first: everything happens in time domain, i.e. we exist in a 4D world, where 3D objects change or move as a function of time.
- And we have our own sensors, which can watch this time evolution: eyes → bandwidth limit: 1 Hz
- For faster or slow processes we develop instruments to capture events and look at them: oscilloscopes, stroboscopes, cameras…

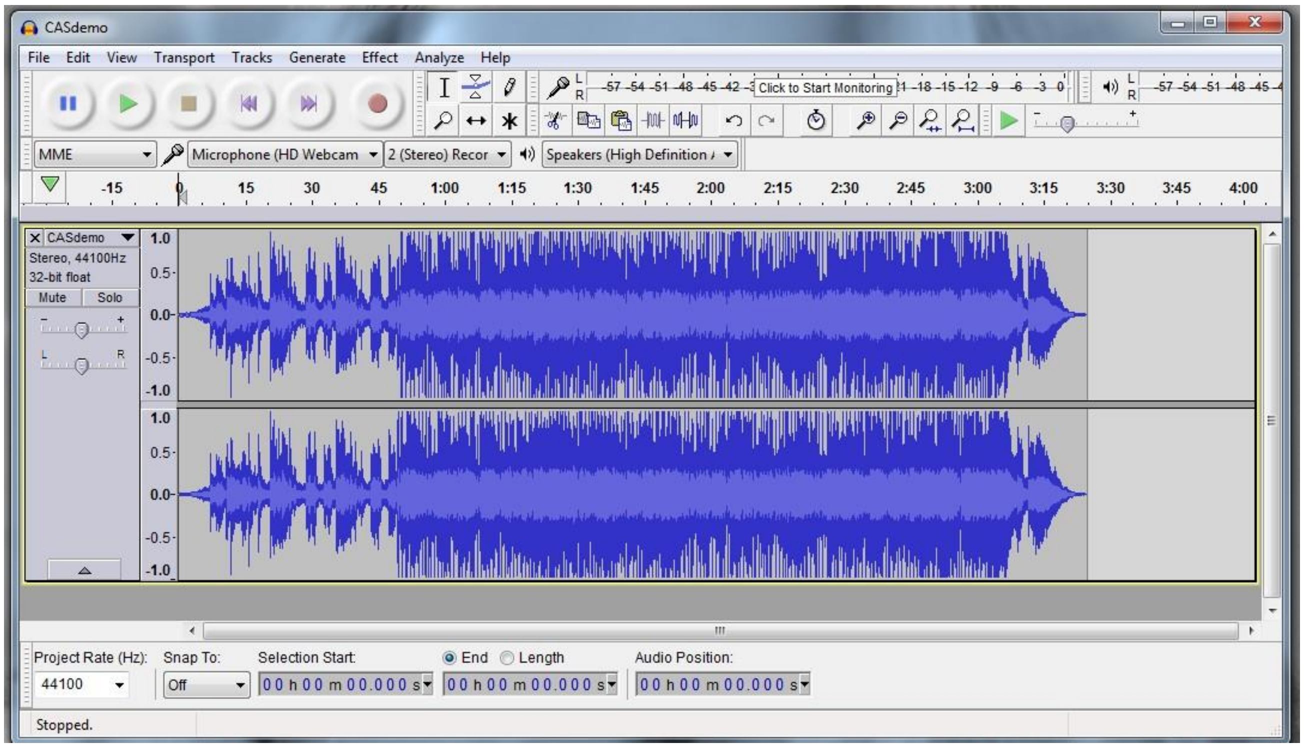


Introduction 2/3

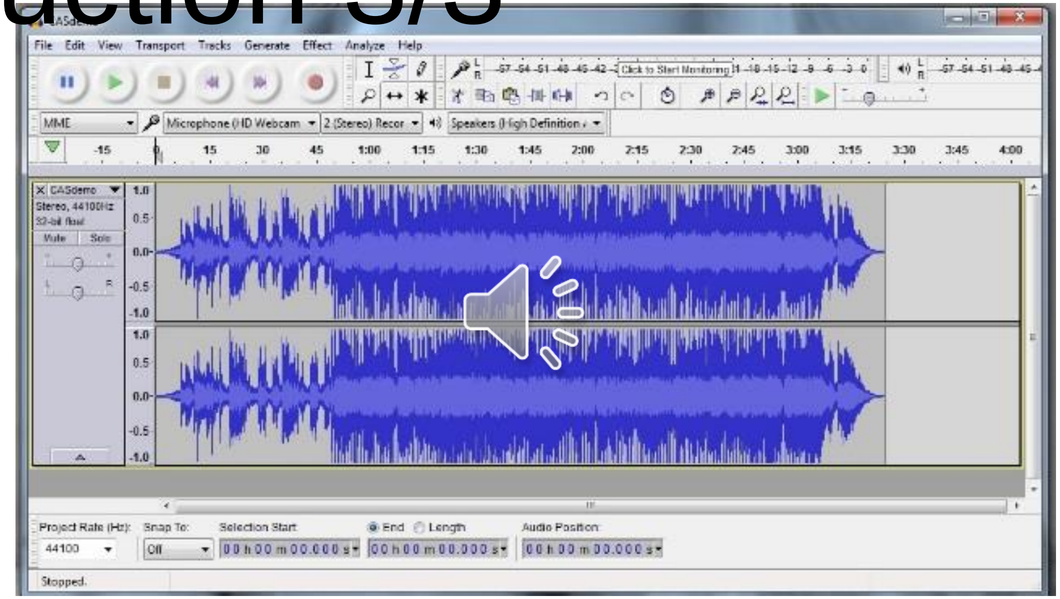
- But we have another sensor: ears



- What is this?



Introduction 3/3



- Once we perceive the material in frequency domain (our brain does this for us), we can better understand the material.

- **Essential:**

Non matter whether we describe a phenomenon in time domain or in frequency domain, we describe the same physical reality. But the proper choice of description improves our understanding!

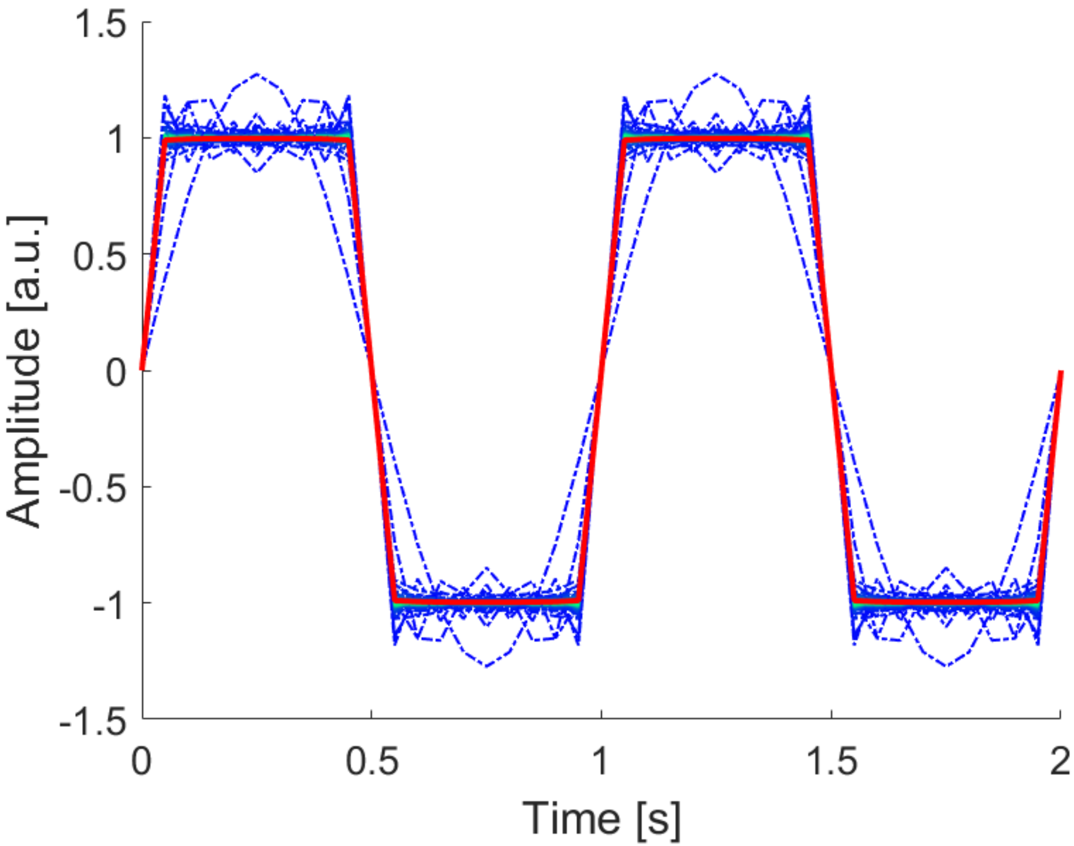
Jean Baptiste Joseph Fourier (1768-1830)

- Had crazy idea (1807):
- **Any** periodic function can be rewritten as a weighted sum of Sines and Cosines of different frequencies.
- Don't believe it?
  - Neither did Lagrange, Laplace, Poisson and other big wigs
  - Not translated into English until 1878!
- But it's true!
  - called Fourier Series
  - Possibly the greatest tool used in Engineering

Fourier Series

Any periodic function *f(x)* can be expressed as a series of harmonics

On the right we see a rectangular periodic Function represented as Sum of the fundamental (a sine wave with the same frequency) and many higher harmonics (odd multiples of the Fundamental) with decreasing amplitudes.



Fourier Transform

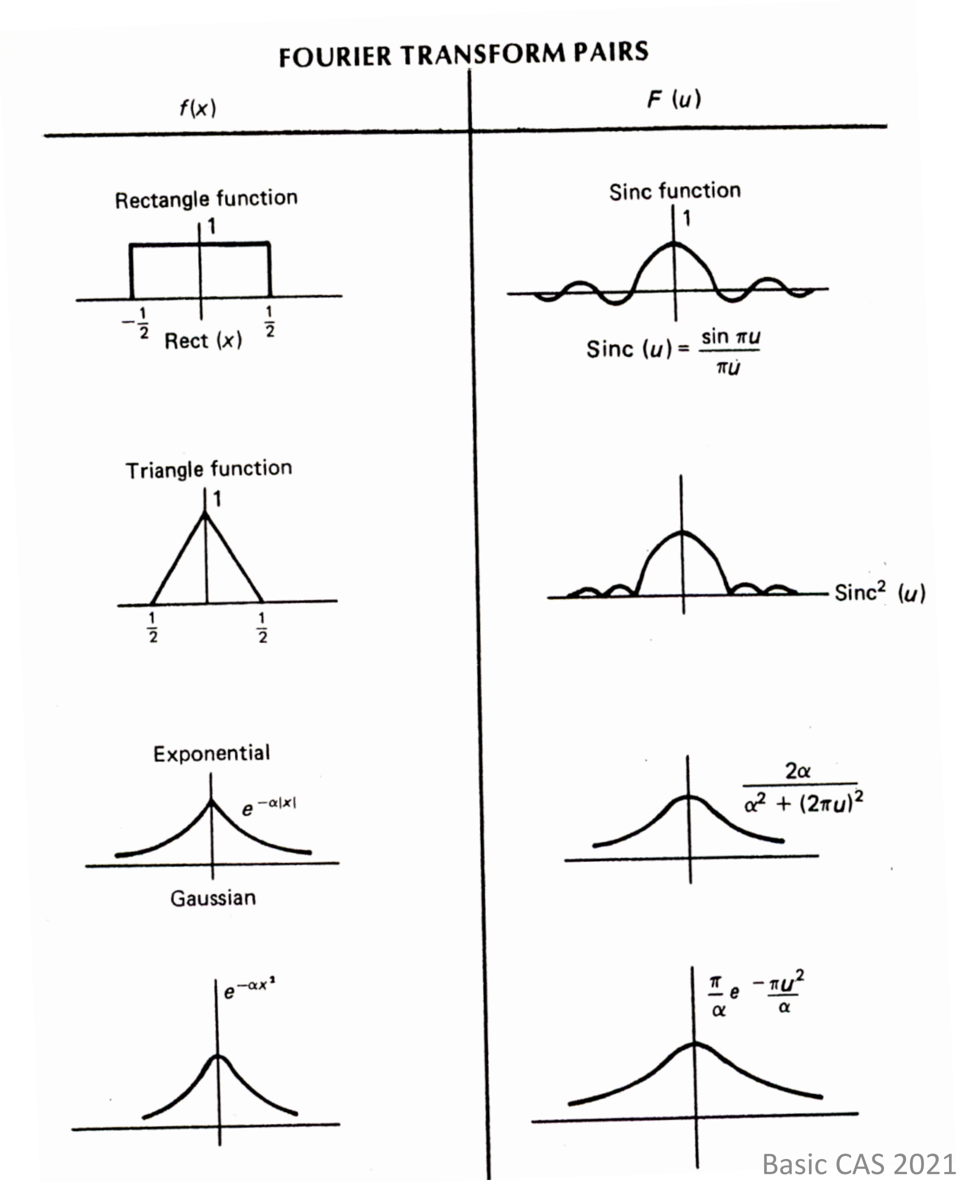
Any non-periodic time-domain function *f(x)* can be transformed by the Fourier-transform (FT) into frequency domain function F(u)

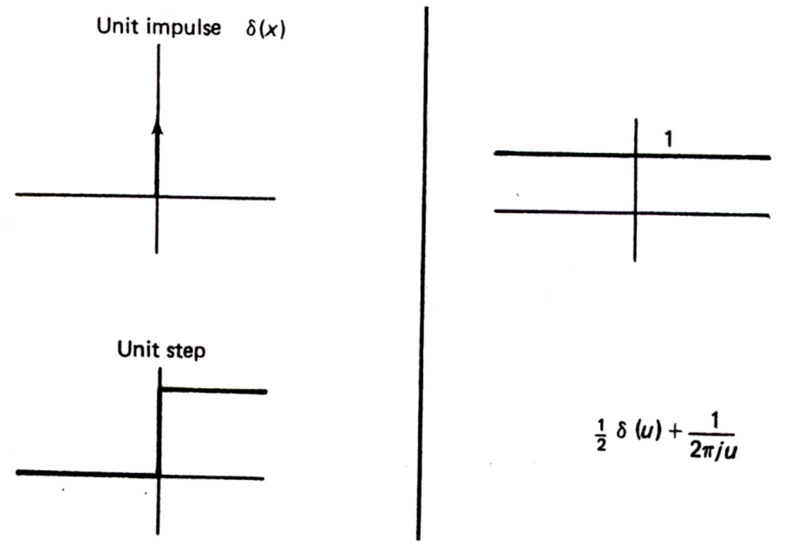
FT defined as:

$$F(u) = \int_{-\infty}^{\infty} f(x) e^{-iux} dx$$

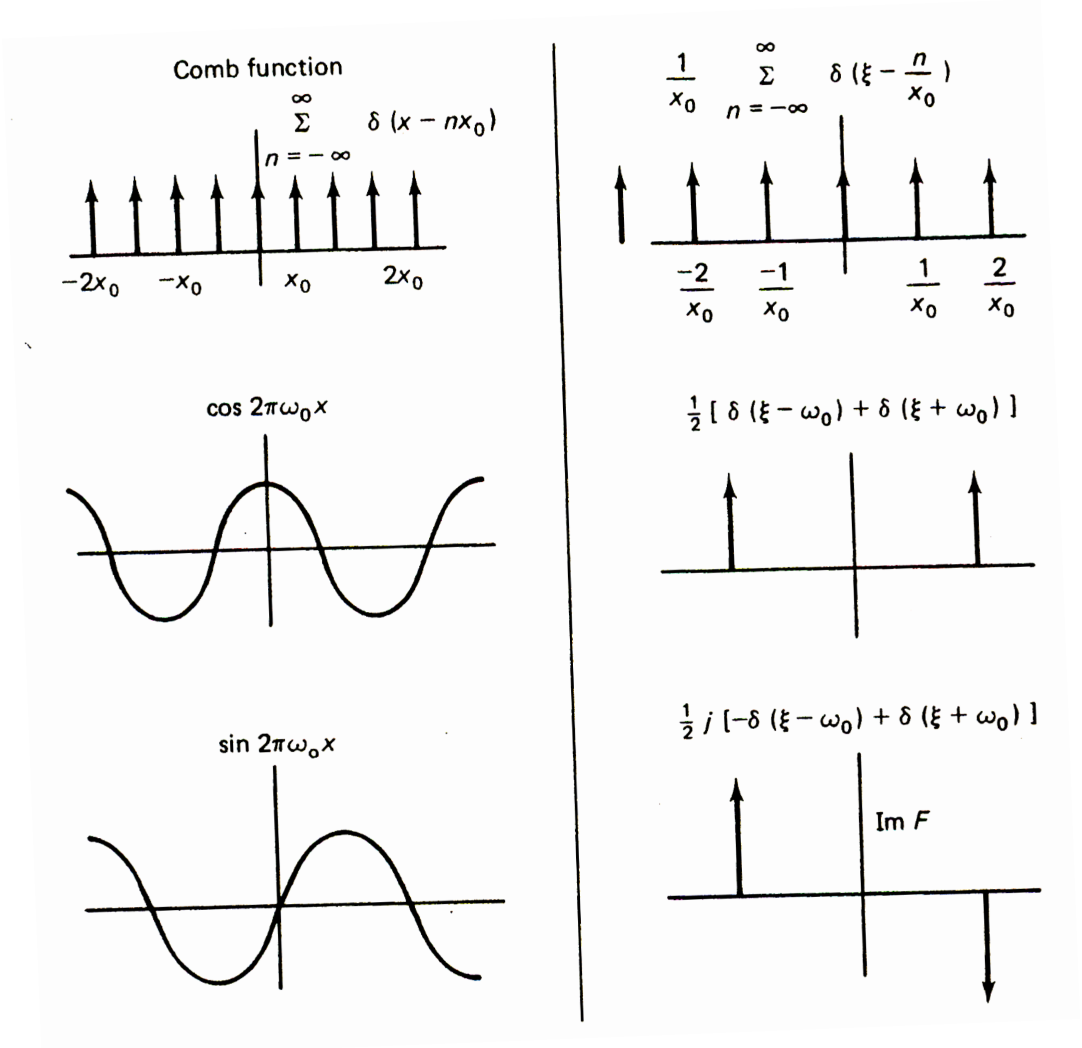
Note: $e^{ik} = \cos k + i \sin k \qquad i = \sqrt{-1}$

Fourier Transform Pairs (I)

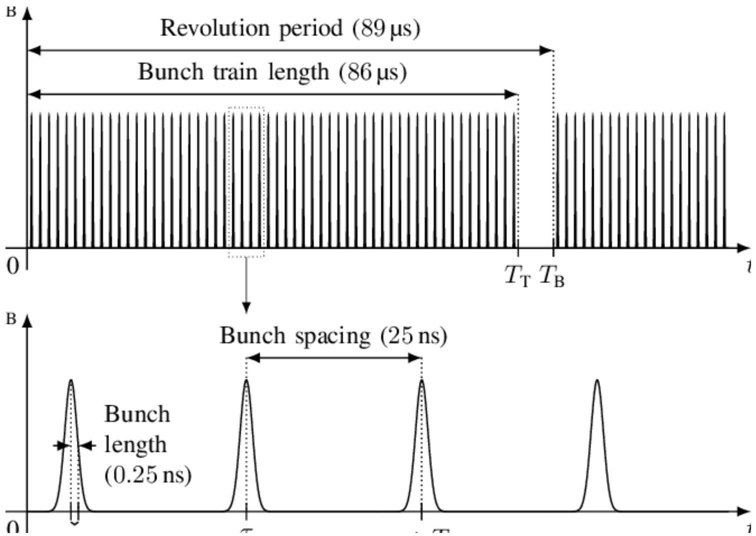




Fourier Transform Pairs (II)



Definitions



In real accelerators not all available RF-buckets are filled with particle bunches.

- a gap must be left for the injection/extraction kickers
- Physics experiments can impose a minimum bunch distance, which is larger than one RF period (i.e. LHC)

Revolution frequency: $\omega_{rev} = 2\pi\ f_{rev}$

RF frequency: $\omega_{RF} = 2\pi\ f_{RF} = h^*\ \omega_{rev}$ (h=harmonic number)

Bunch Repetition frequency: $\omega_{rep} = 2\pi\ f_{rep} = \omega_{rev}\ /n$ (n= number of RF buckets between bunches)
(f_{rep} = 1/bunch spacing)

Nominal LHC Filling Scheme

"Standard Filling Schemes for Various LHC Operation Modes", R. Bailey and P. Collier, **LHC-Project-note-323**.

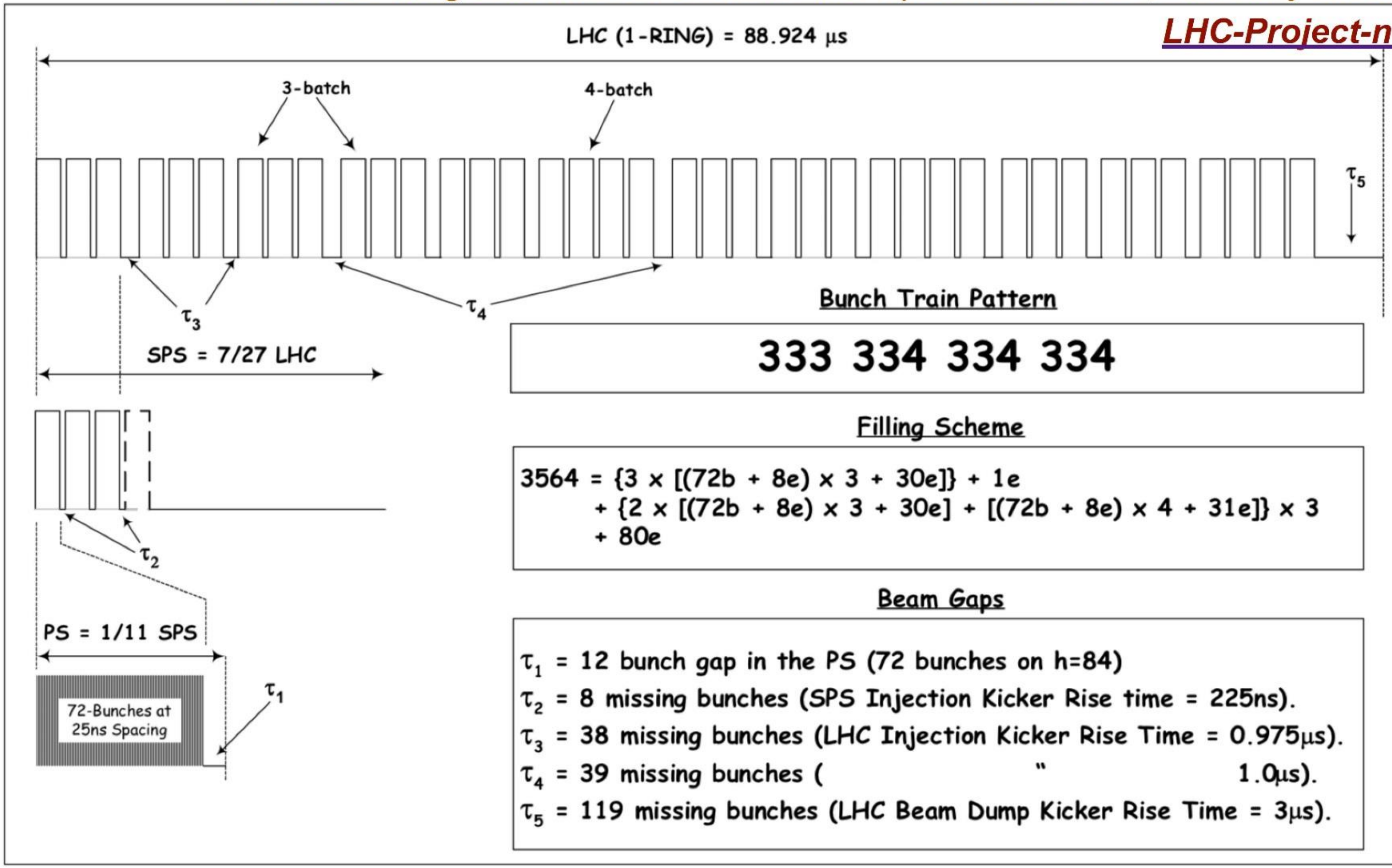


Figure 1: Schematic of the Bunch Disposition around an LHC Ring for the 25ns Filling Scheme

Understanding beam signals in time and frequency domain

We start with:

Single bunch single pass

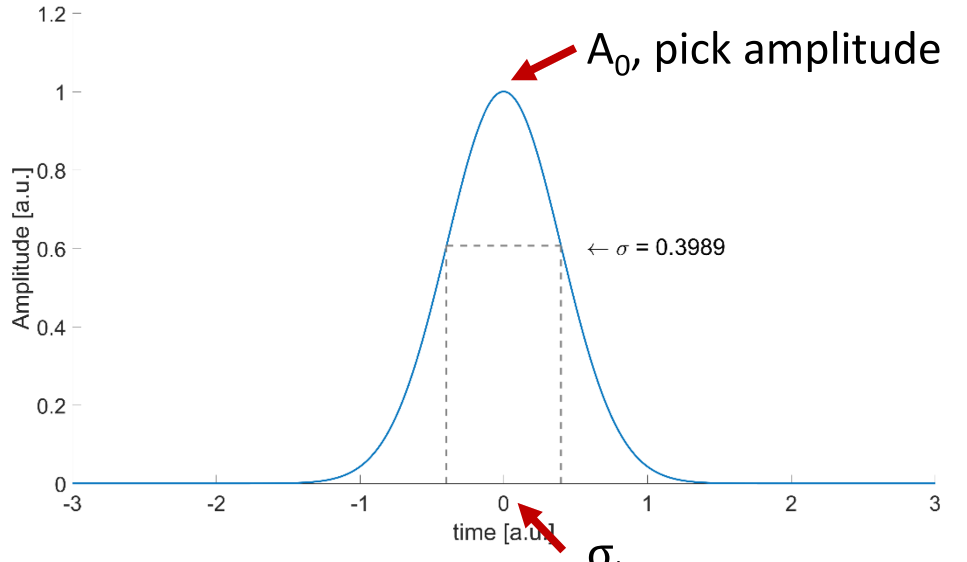
- Time and frequency domain description
- Measurement of bunch length in time domain
- Measurement of bunch length in frequency domain

Particle beam with gaussian longitudinal distribution

Time domain

$$f(t) = A_0 \exp\left(-\frac{t^2}{2\sigma_t^2}\right)$$

$$area = \int_{-\infty}^{+\infty} f(t)dt = \sqrt{2\pi}A_0\sigma_t$$

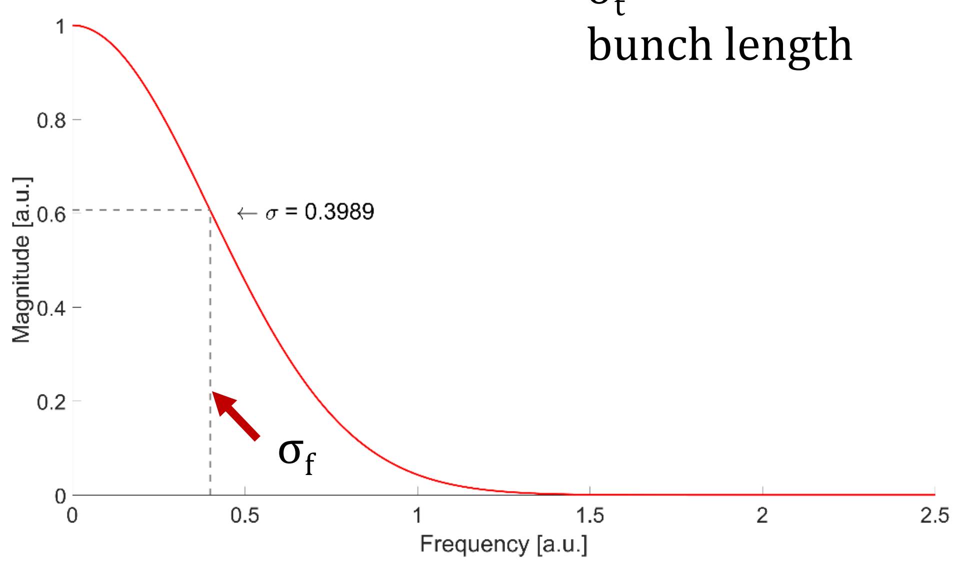


Frequency domain

$$F(k) = \frac{A_0}{\sqrt{2\pi}\sigma_f} \exp\left(-\frac{k^2}{2\sigma_f^2}\right)$$

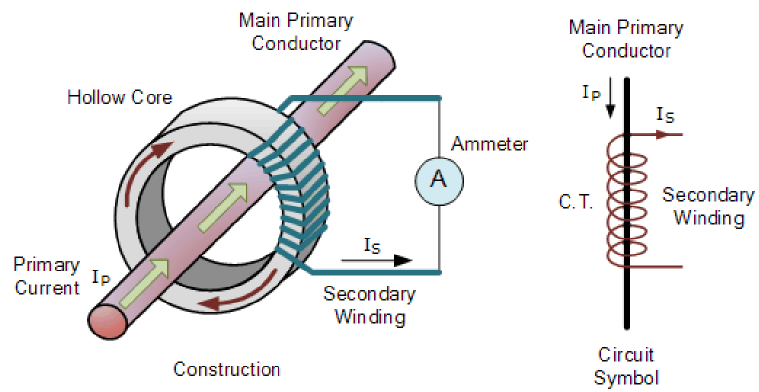
$$\sigma_f = \frac{1}{2\pi\sigma_t}$$

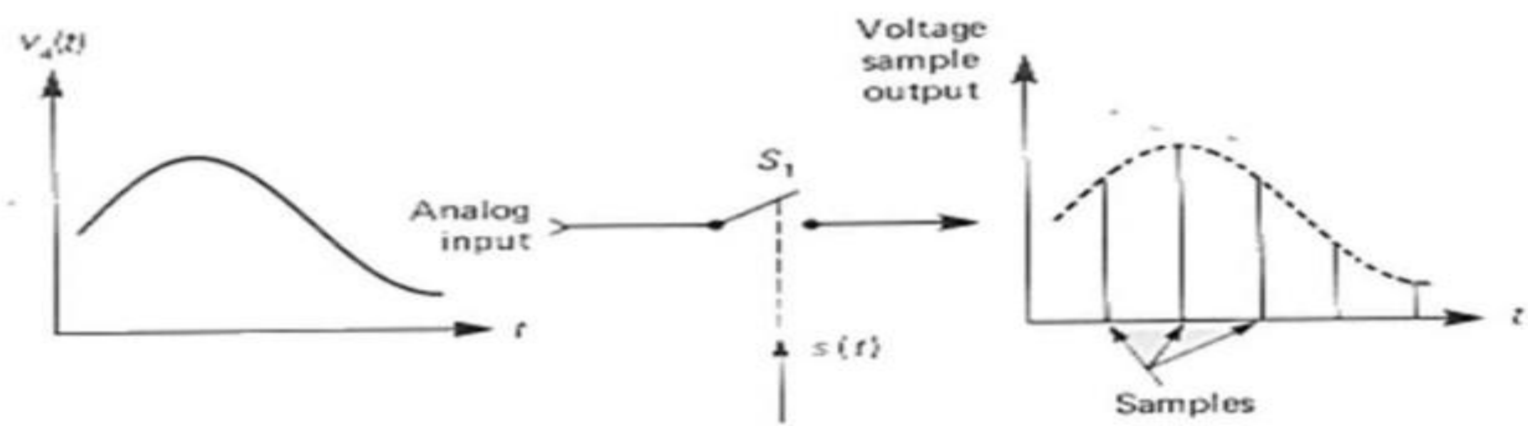
$$F(0) = area = \frac{A_0}{\sqrt{2\pi}\sigma_f} = \sqrt{2\pi}A_0\sigma_t$$



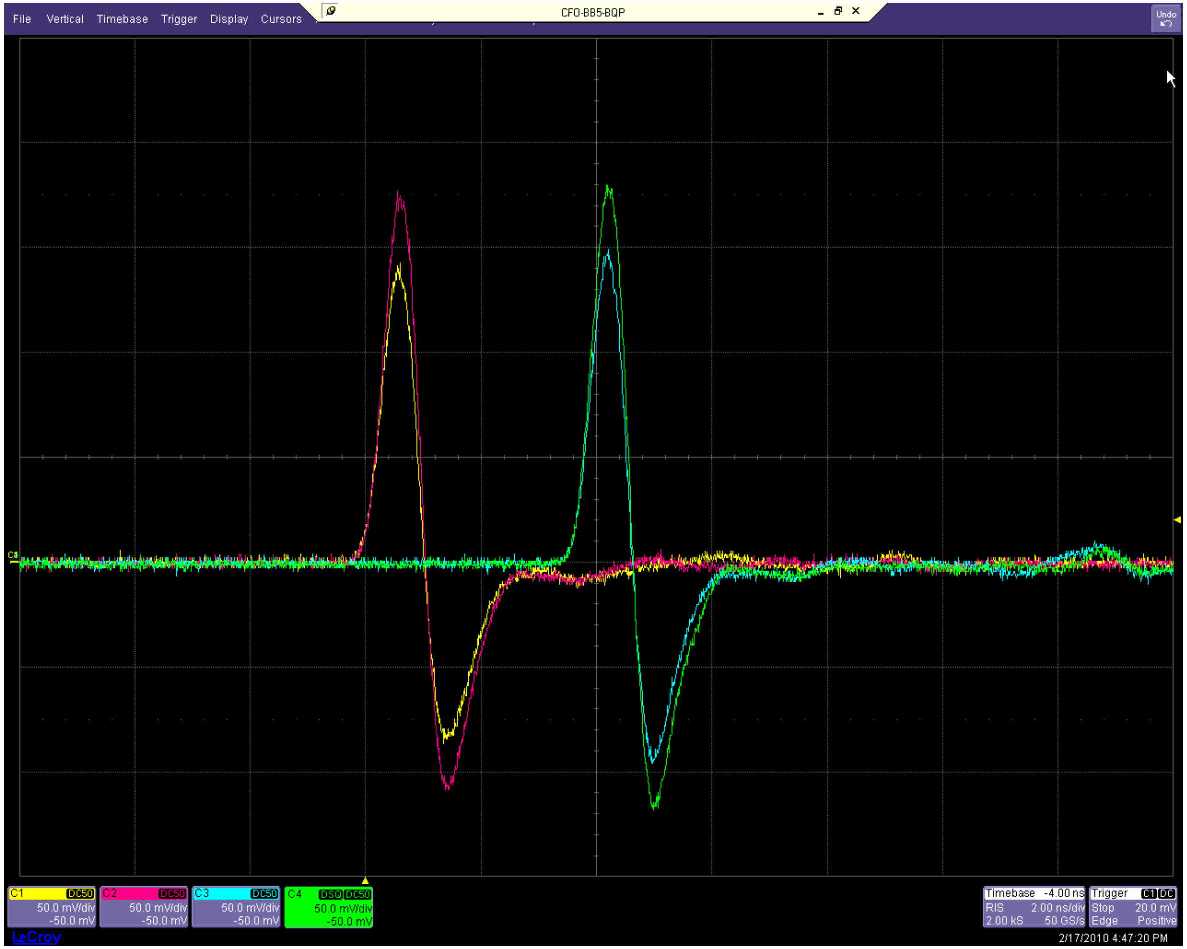
Time domain measurement of single bunch

- Sampling (=measurement) of an electrical signal in regular time intervals. The electrical signal is obtained from a monitor, which is sensitive to the particle intensity.



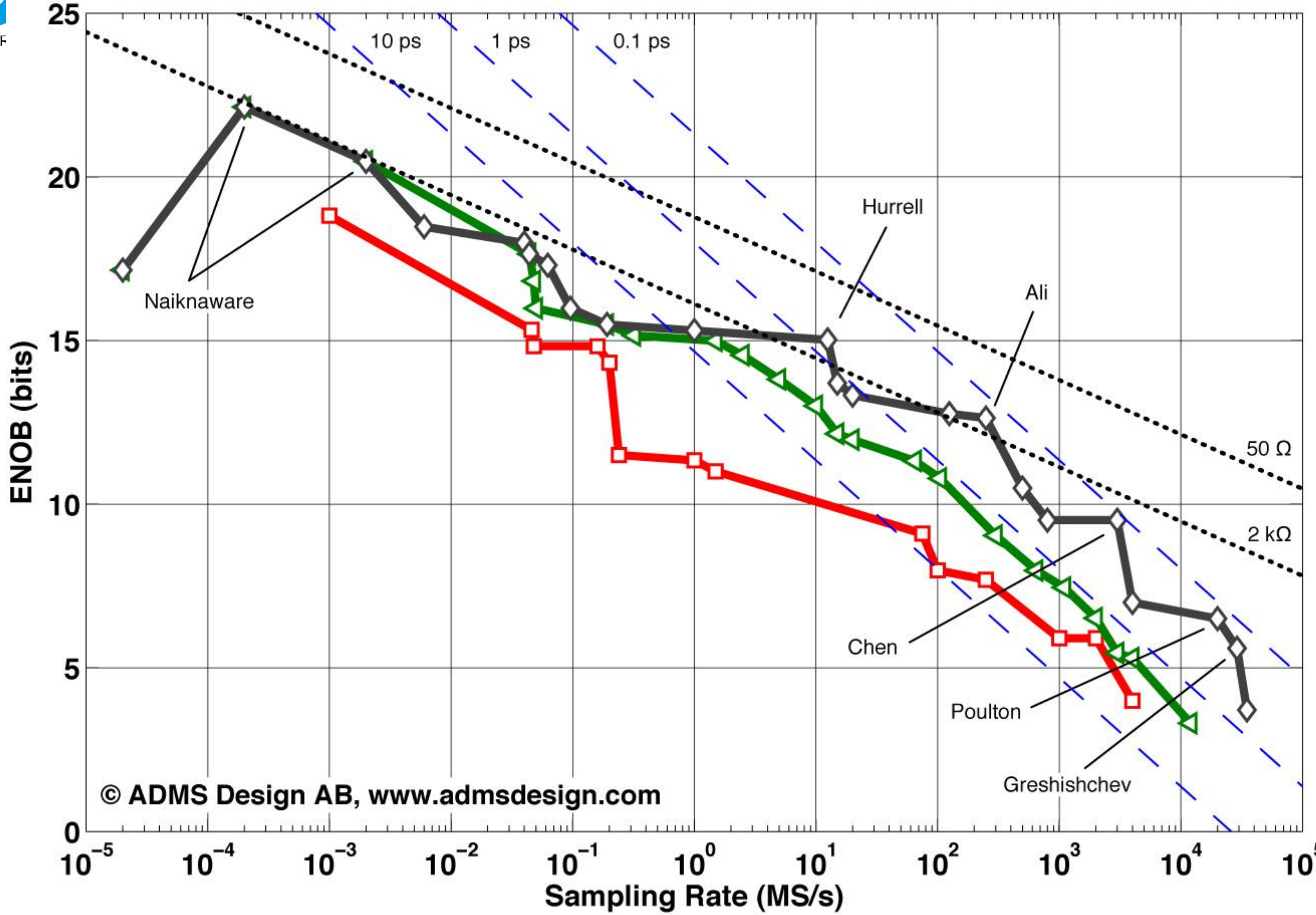


Sampling a pulse at 2 Gigasamples/sec



- 50 mV/div, 2 ns/div
- SPS beam
- 2 pairs of 10 mm button electrodes (second pair delayed by cables for clarity)
- Signals already “filtered” by quite long cables

ADC performance chart (2019)



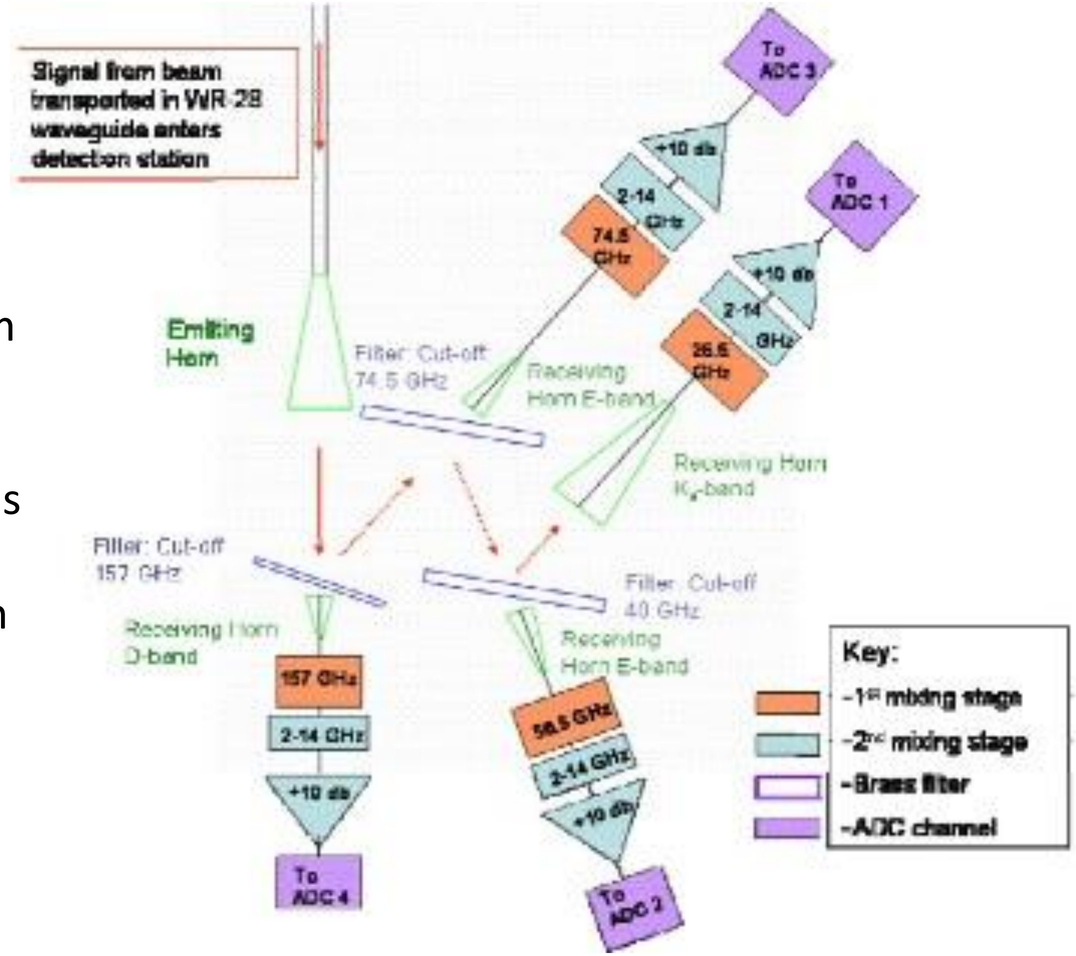
Frequency domain measurement of single bunch

Nice example from R&D work in CTF3 (CERN)
A.Dabrowski et al., Proc of PAC07, FRPMS045

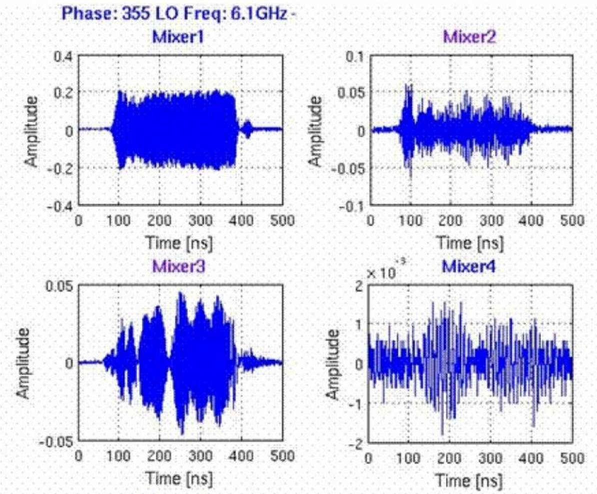
Primary signal is EM wave of beam extracted through a thin window

Subdivision into 4 frequency bands

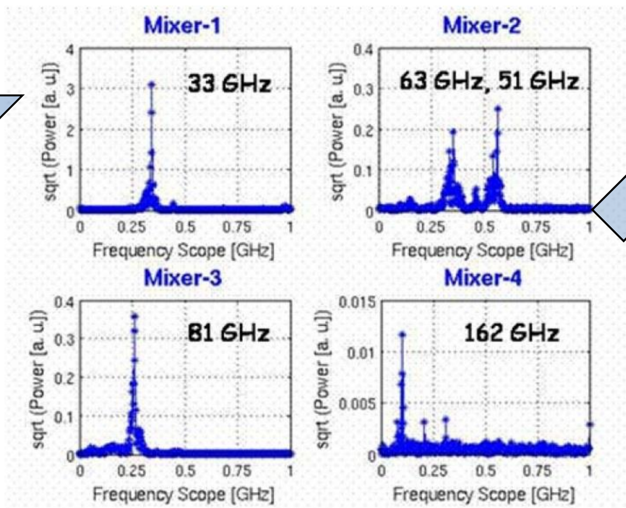
Measurement of rms amplitude in the 4 bands



CTF3 results



Time domain measurements of 4 bands



FFT of down-converted signals

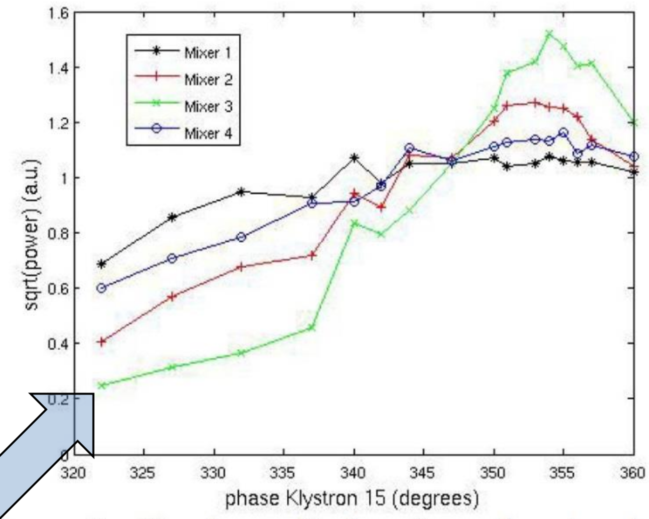


Figure 5: Signal amplitudes from the 4 selected frequencies as a function of the phase in Klystron 15.

fit

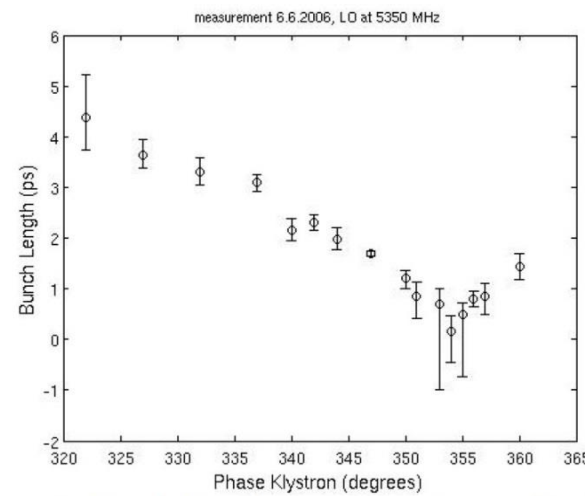


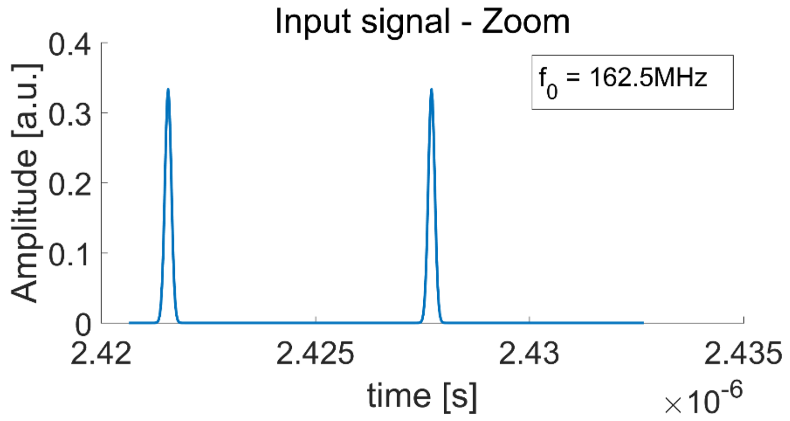
Figure 6: Bunch length measurements as a function of the phase of Klystron 15

Single bunch multi pass (circular accelerator) → "Revolution harmonics"

Time domain

$$f(t) = \sum_{n=1}^{N} A_0 \exp\left(-\frac{(t-nT)^2}{2\sigma_t^2}\right)$$

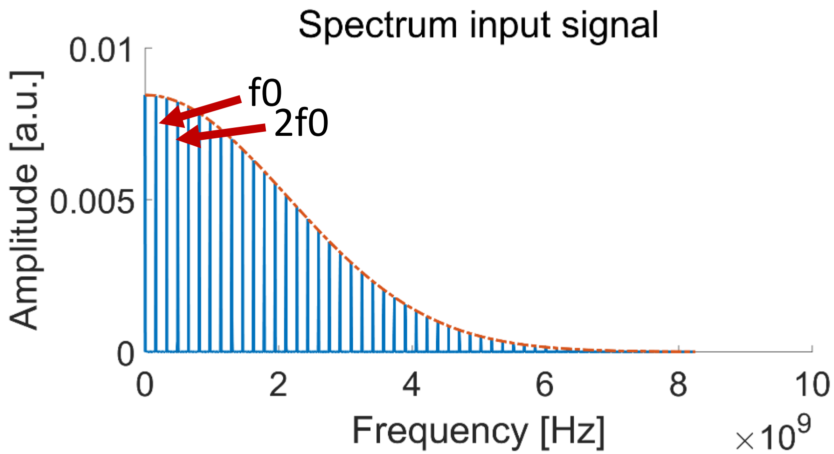
$$area = \int_{-\infty}^{+\infty} f(t)dt = N \times \sqrt{2\pi} A_0 \sigma_t$$



Frequency domain

$$F(k) = \sum_{i=1}^{N} F_c(ik_0) \exp\left(-\frac{(k-ik_0)^2}{2\sigma_f^2}\right),$$

$$\sigma_f = \frac{1}{2\pi\sigma_t}$$



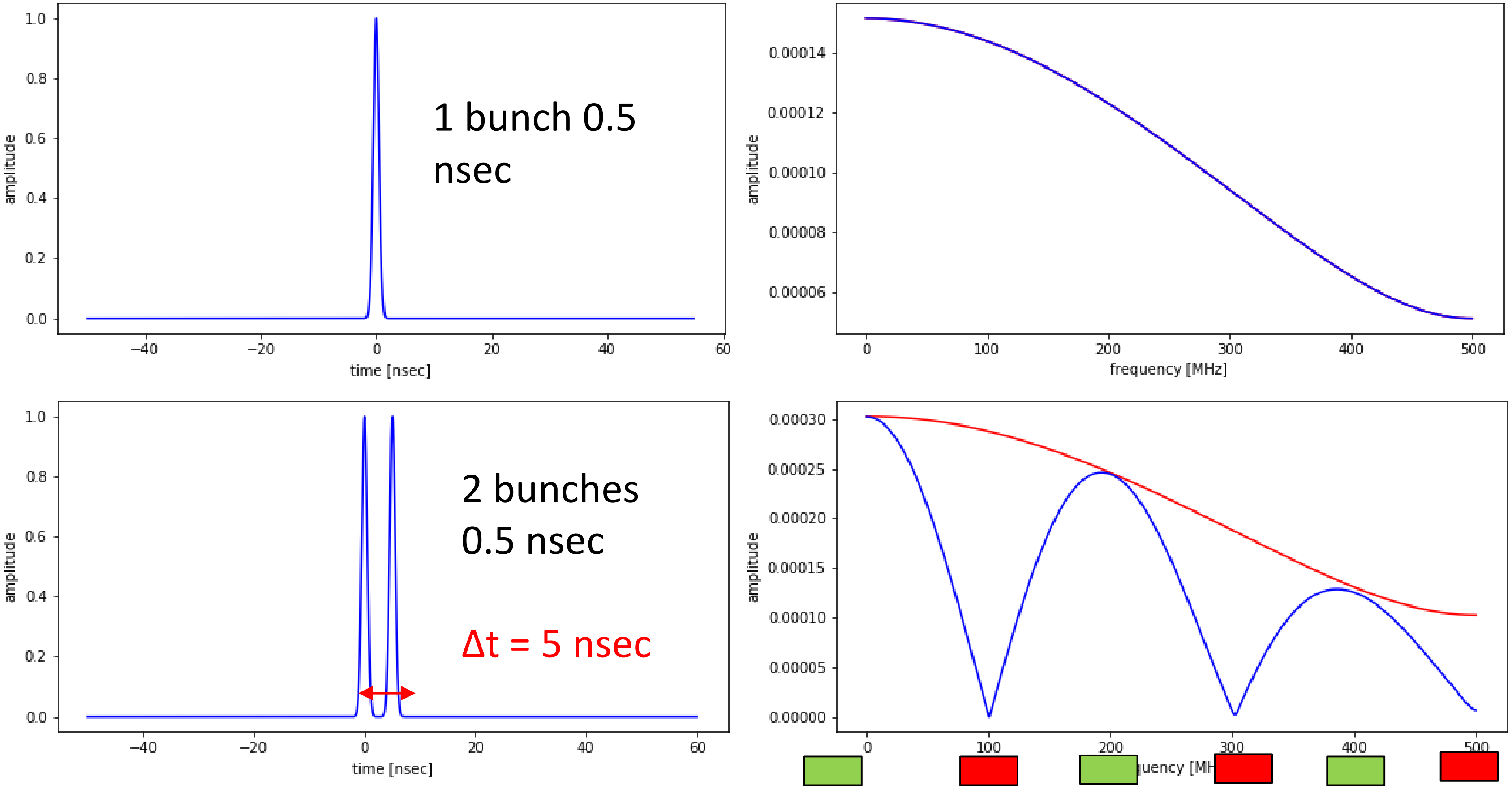
Understanding the revolution harmonics

- The continuous spectrum of a single bunch passage becomes a line spectrum.
- The line spacing is $f_{rev}= 1/T_{rev}$. (T_{rev} = revolution time)
- The amplitude envelope of the line spectrum is the “old” single pass frequency domain envelope of the single bunch.

- Why?
  - short answer: Do the Fourier transform!
  - long answer:

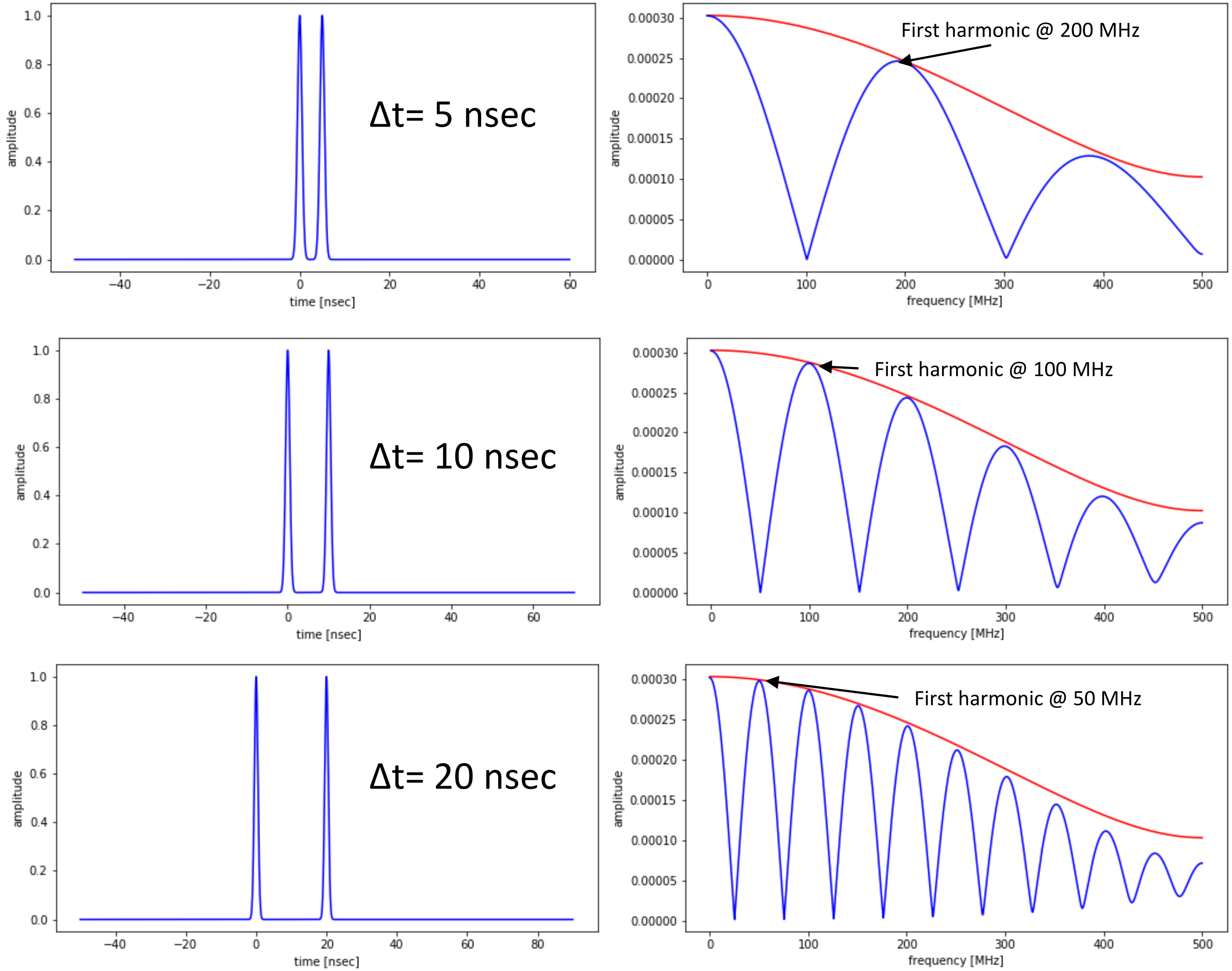
  Understand in more detail 2,3,4…N consecutive bunch passages in time and frequency domain (next slides)

Bunch pattern simulations (1/4)

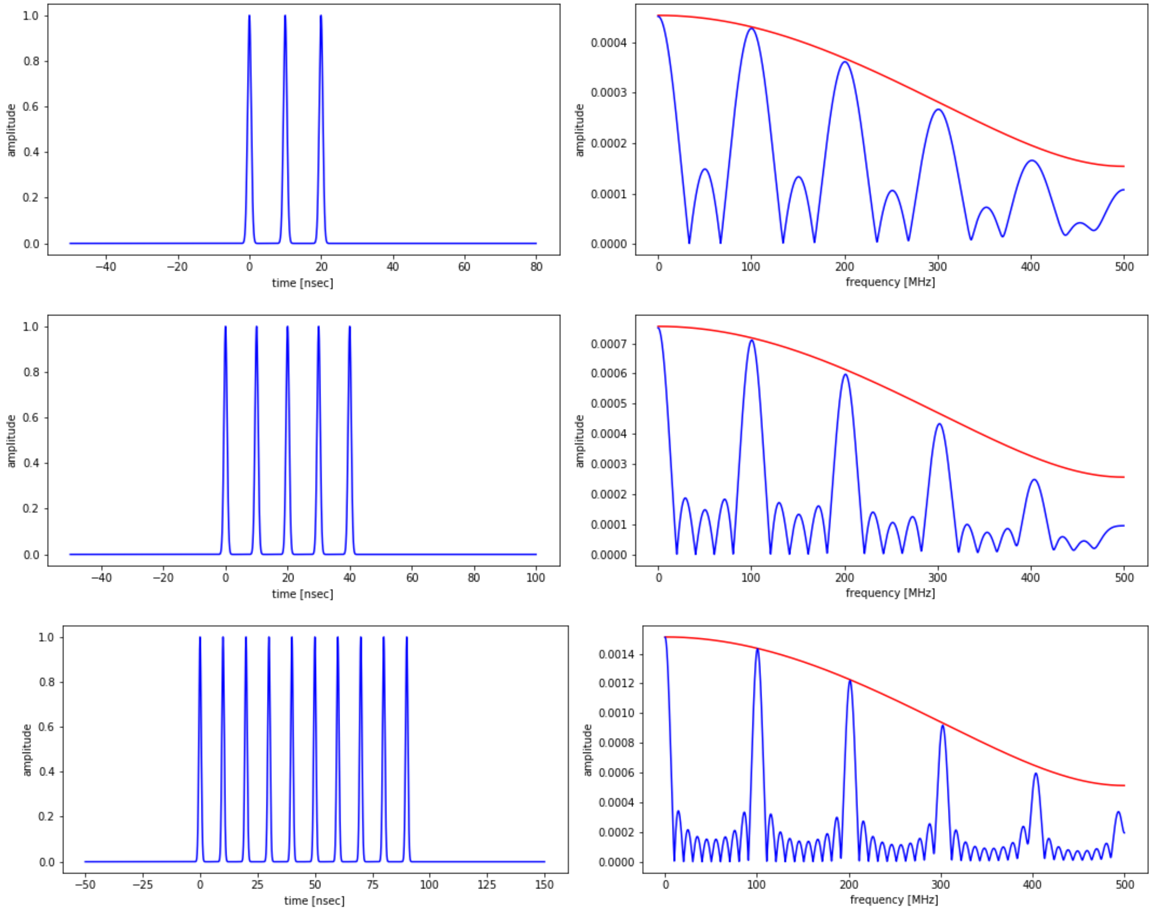


- Frequencies in this range make a constructive interference (no phase difference)
- Frequencies in this range cancel each other (180^0 phase difference)
- Other frequencies intermediate summation/cancelation

Bunch pattern simulations (2/4)

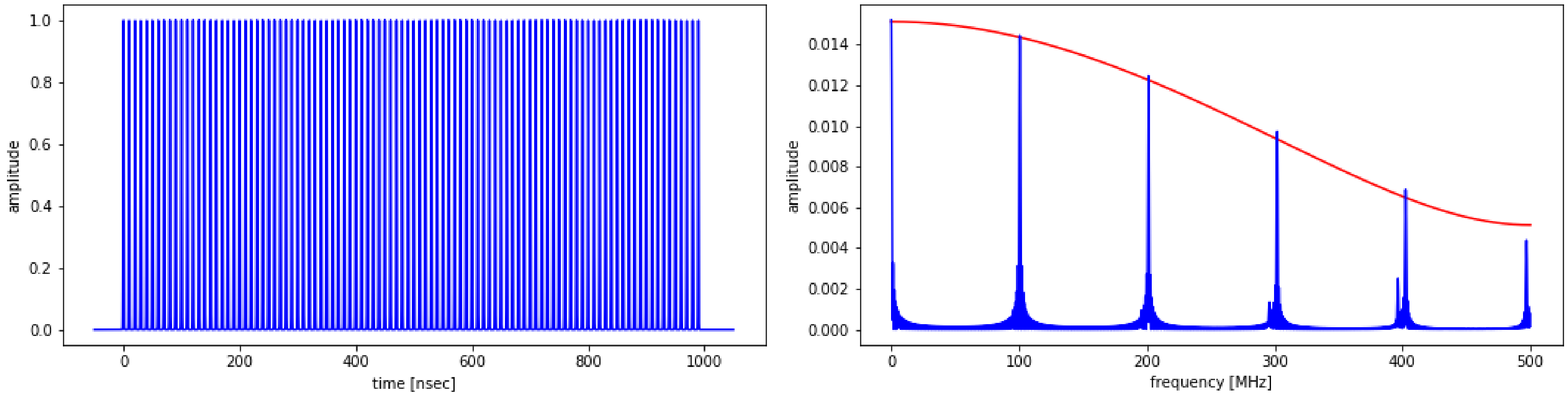


Bunch pattern simulations (3/4)



From top to bottom:
3, 5, 10 bunches (0.5nsec long, Δt = 10 nsec)

Last bunch pattern simulation



- 100 equidistant bunches (Δt = 10 nsec)
- Resulting spectrum is a line spectrum with the fundamental line given by the inverse of the bunch distance

A Measured Longitudinal beam spectrum

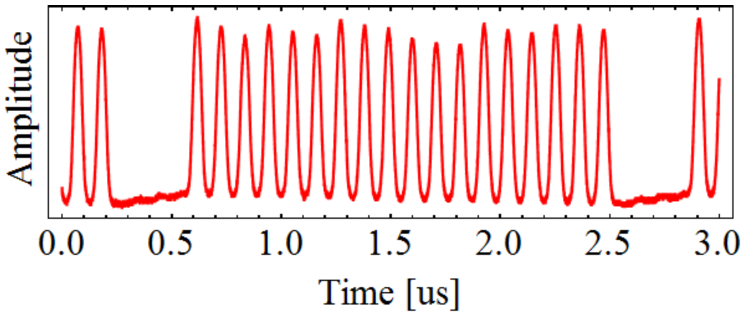
- **Circular accelerator**
  - → **Beam signal periodic with revolution frequency:** ω_{rev}

→ **Spectral components at:**

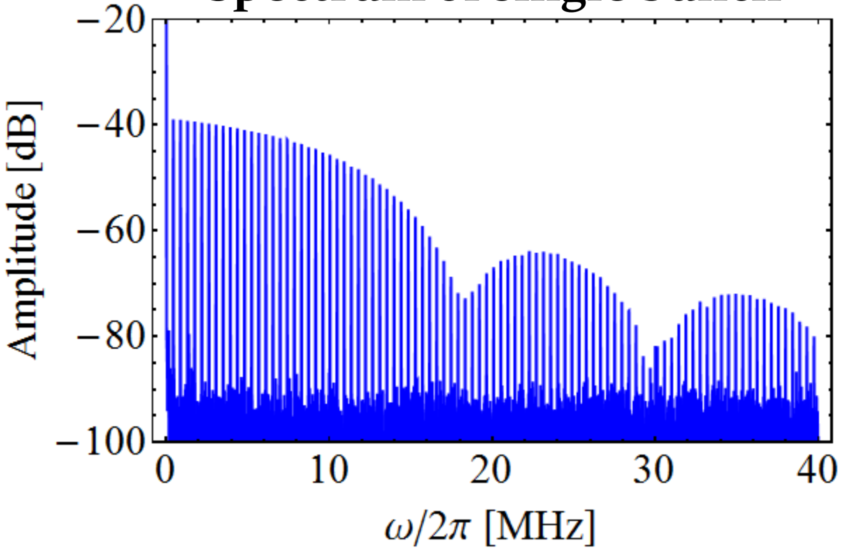
$$\omega = n\omega_{\mathrm{rev}}$$

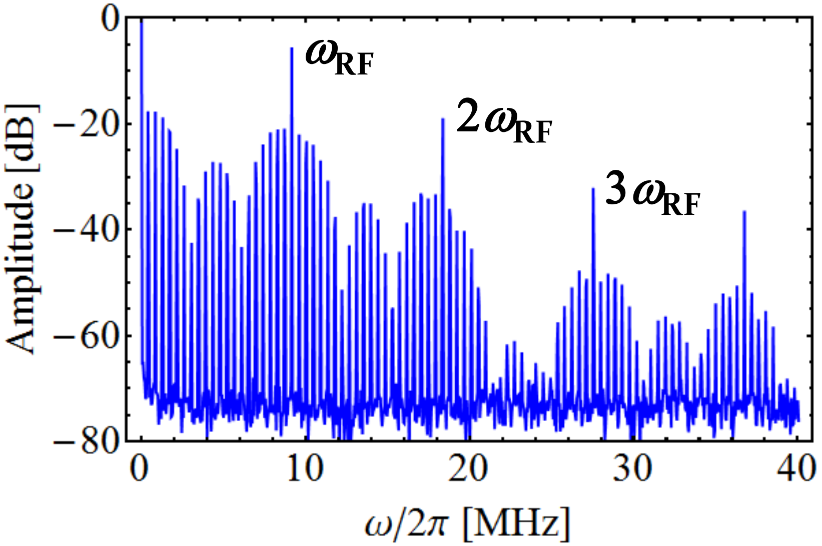
Bunch not Gaussian.
Somewhat between triangular and parabolic

Multi-bunch beam

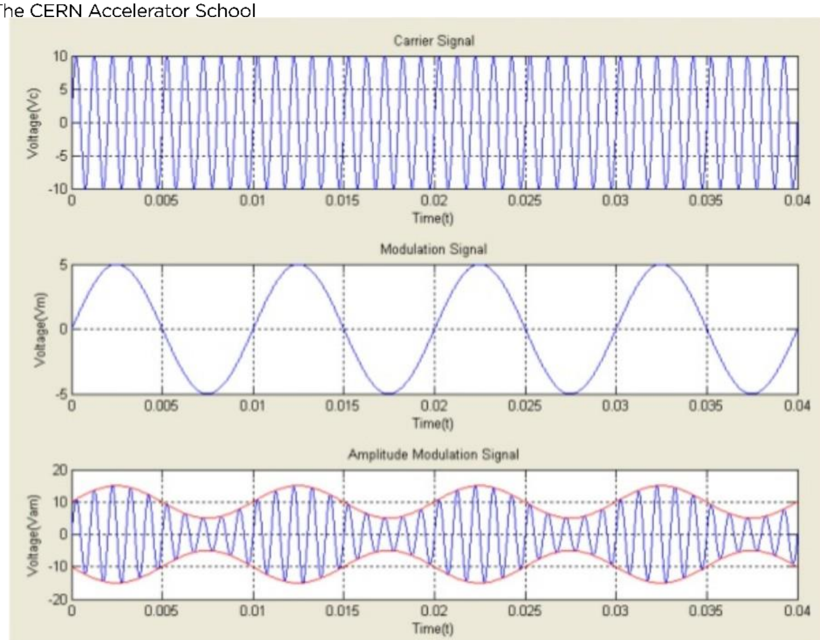


Spectrum of single bunch





Amplitude modulation



$$v = V_{env} \sin 2\pi f_c t$$
$$= V_c (1 + m \sin 2\pi f_m t) \bullet \sin 2\pi f_c t$$

m= modulation index 0…1 ($V_{env} = V_c$)

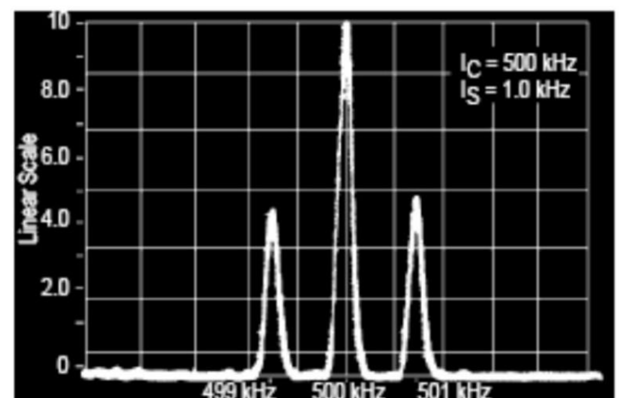
Using trigonometric identity:

$$(\sin a)(\sin b) = 1/2[\cos(a-b) - \cos(a+b)]$$

$$v = V_c \sin 2\pi f_c t$$
$$+ \frac{m}{2} V_c \cos 2\pi (f_c - f_m) t$$
$$- \frac{m}{2} V_c \cos 2\pi (f_c + f_m) t$$

$$v_{AM} = V_c \sin 2\pi f_c t + \frac{V_m}{2} \cos 2\pi (f_c - f_m) t - \frac{V_m}{2} \cos 2\pi (f_c + f_m) t$$

Carrier LSB USB



Relevant example of amplitude modulation: stimulated betatron oscillation(or: tune measurement)

taken from R.Jones, proc. of BI-CAS 2018

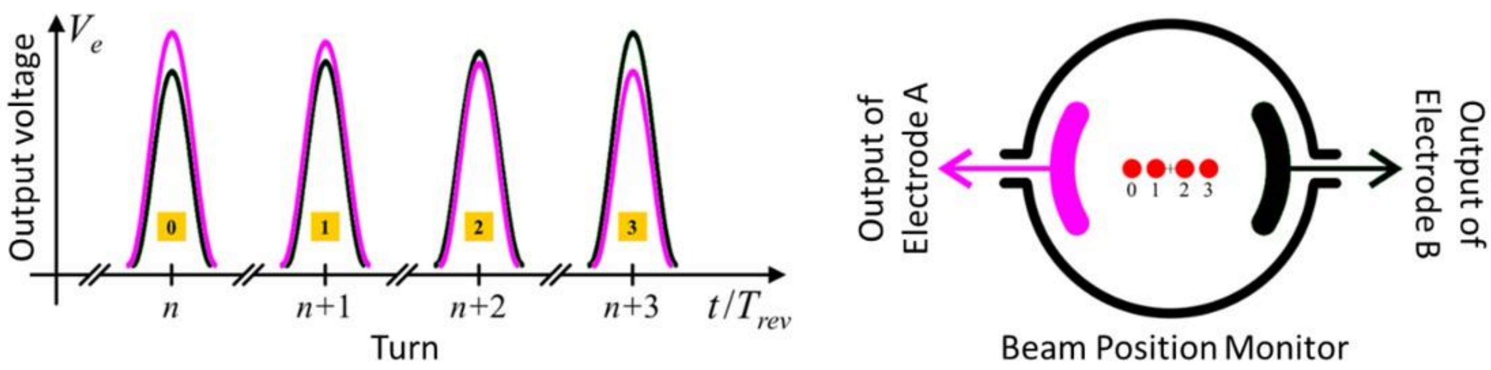


Fig. 4: Detecting oscillations using a beam position monitor. The oscillation information is superimposed as a small modulation on a large intensity signal.

Beam centre of charge makes small betatron oscillation around the closed orbit
(- stimulated by an exciter or by a beam instability)

Depending on the proximity to an EM sensor the measured signal amplitude varies.

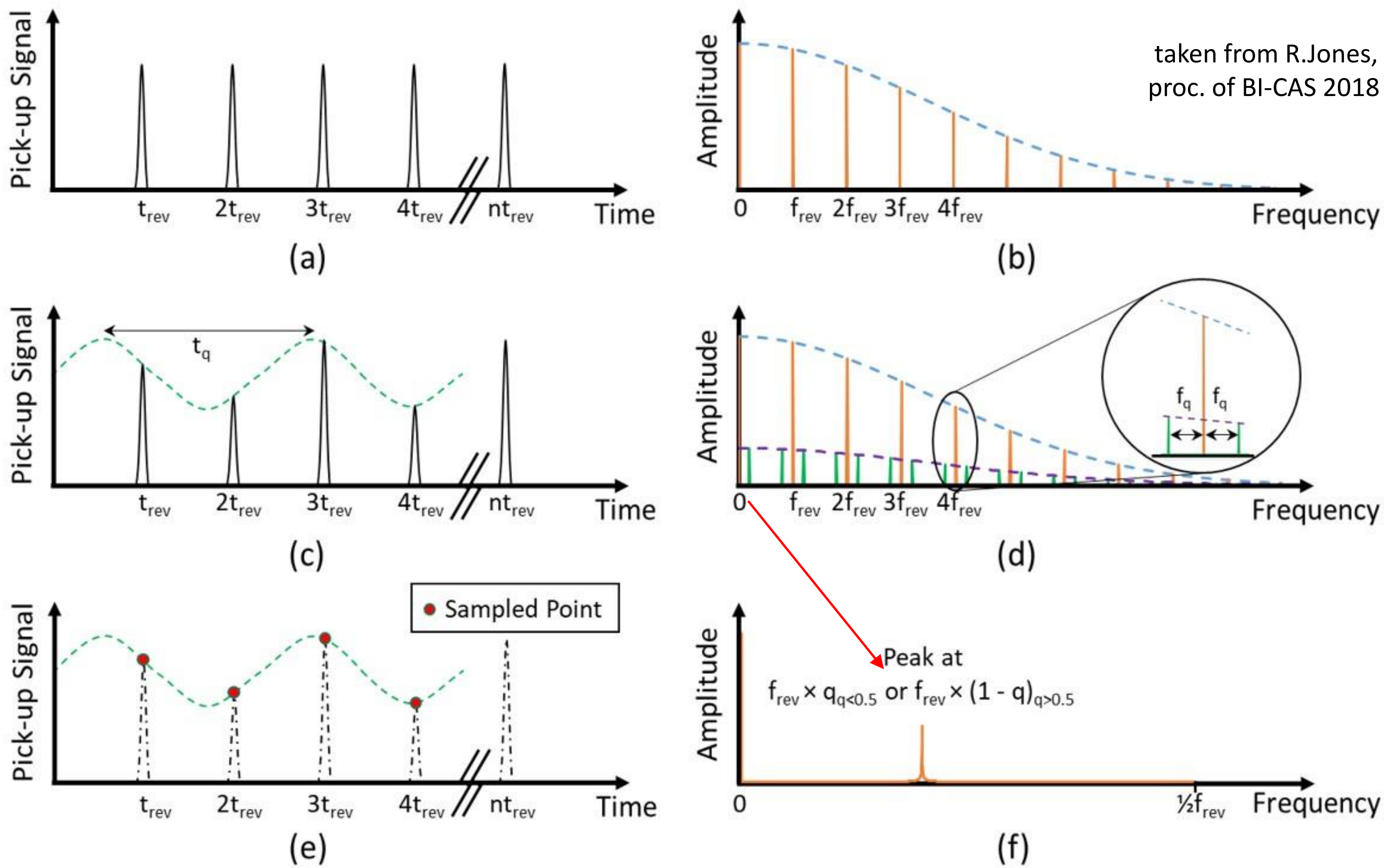


Fig. 2: Time and frequency domain representation for a bunch of particles observed at one single location on the circumference of the accelerator. (a & b) continuous measurement without betatron oscillation; (c & d) continuous measurement undergoing betatron oscillation (50% modulation); (e & f) sampled once per revolution.

Discrete Fourier Transforms

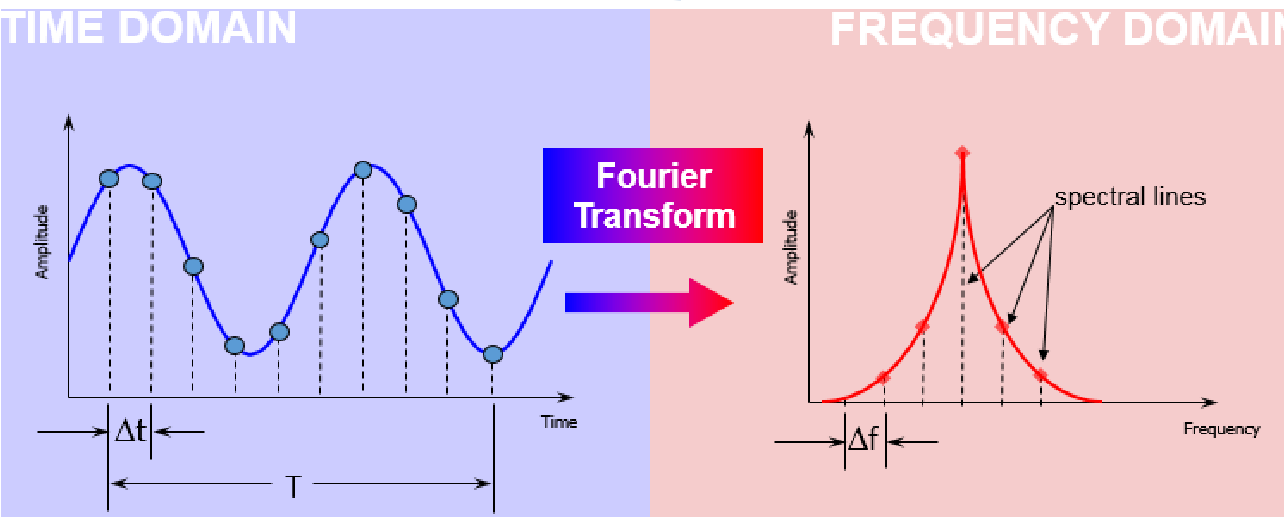
- Discrete Fourier Transform basics

In general:

We use DFTs of N equidistant time sampled signals;

A FFT (Fast Fourier transform) is a DFT with $N = 2^k$

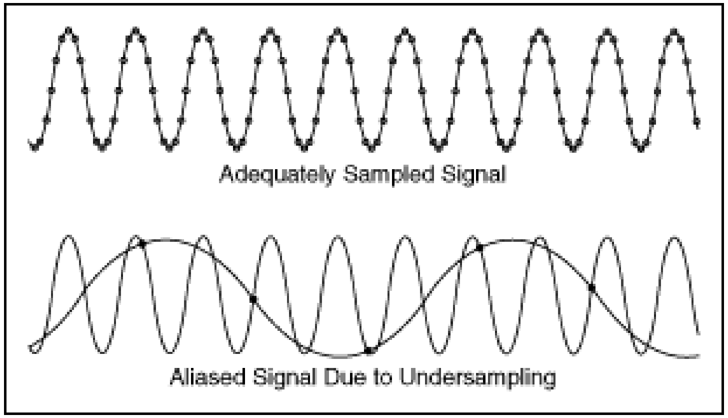
Time Duration		
Finite	Infinite	
Discrete FT (DFT) $X(k) = \sum_{n=0}^{N-1} x(n) e^{-j\omega_k n}$ $k = 0, 1, \ldots, N-1$	Discrete Time FT (DTFT) $X(\omega) = \sum_{n=-\infty}^{+\infty} x(n) e^{-j\omega n}$ $\omega \in (-\pi, +\pi)$	discr. time n
Fourier Series (FS) $X(k) = \int_0^P x(t) e^{-j\omega_k t} dt$ $k = -\infty, \ldots, +\infty$	Fourier Transform (FT) $X(\omega) = \int_{-\infty}^{+\infty} x(t) e^{-j\omega t} dt$ $\omega \in (-\infty, +\infty)$	cont. time t
discrete freq. k	continuous freq. ω	



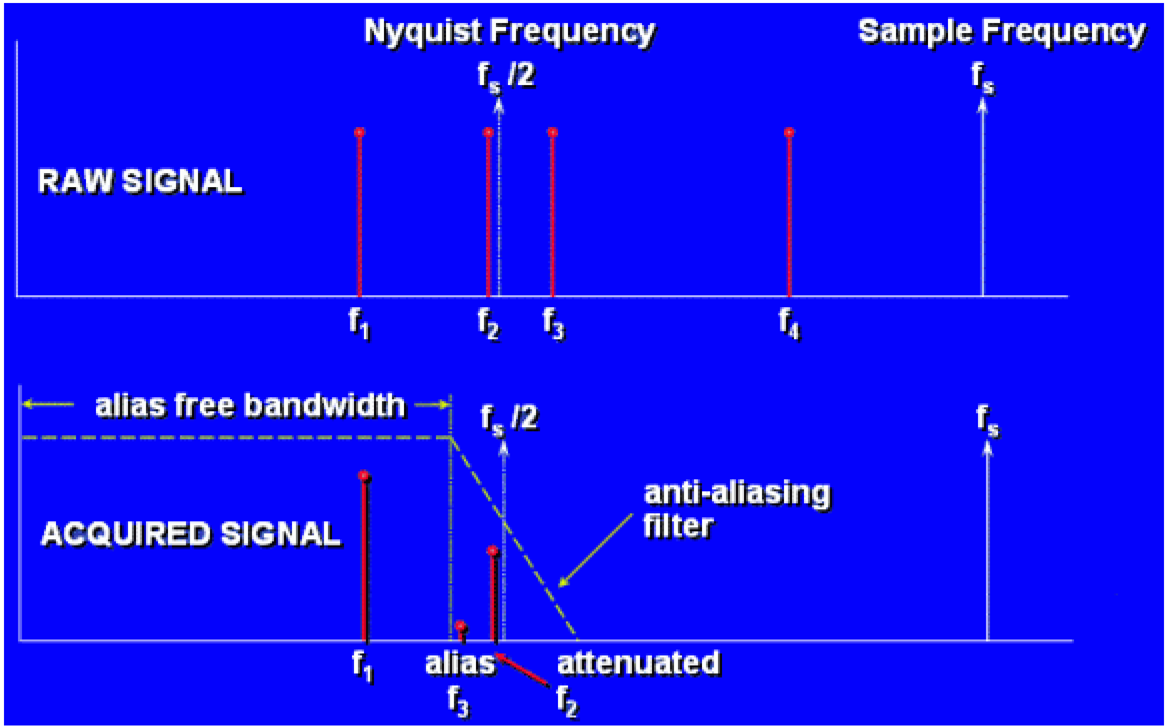
Sampling rate (samples/sec)	$F_s = 1/\Delta t$
Frame Size (seconds)	$T = N \Delta t$
Block Size (# samples)	N

Bandwidth or Max Freq (Hz)	$F_{max} = F_s/2$
Frequency Resolution (Hz)	$\Delta f = F_{max} / SL$
Spectral Lines (# samples)	$SL = N/2$

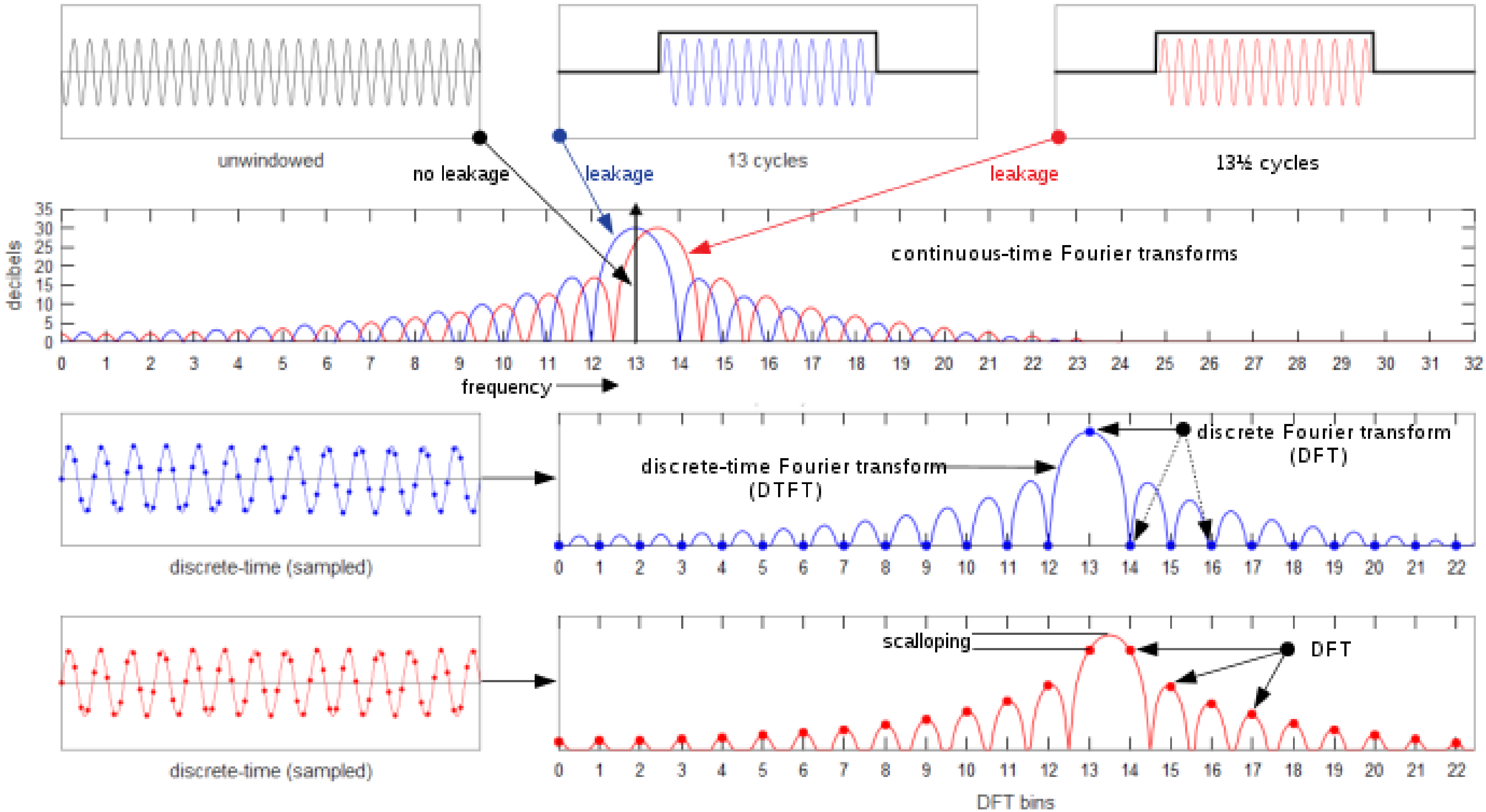
DFT - aliasing



- Periodic signals, which are sampled with at least 2 samples per period, can be unambiguously reconstructed from the frequency spectrum. (Nyquist-Shannon Theorem)
- In other words, with a DFT one only obtains useful information up to half the sampling frequency.
- Antialiasing filters before the sampling suppress usually unwanted higher spectral information.



Spectral leakage caused by windowing

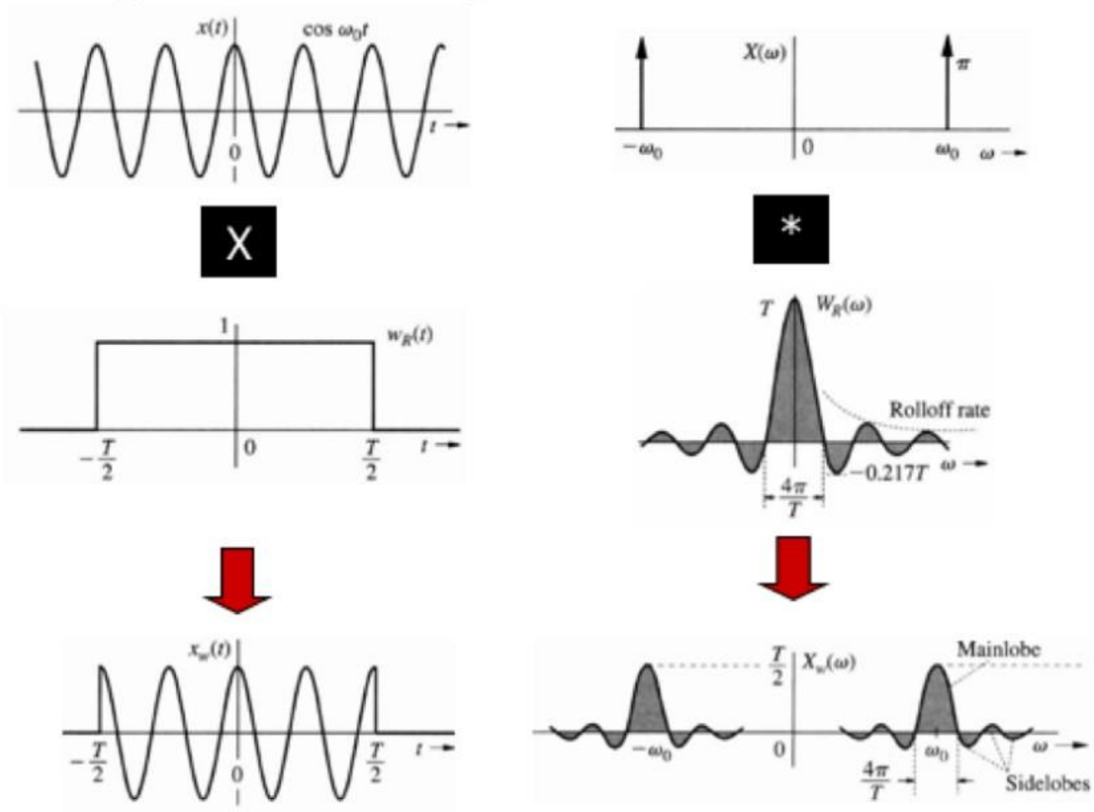


By measuring a continuous signal only over a finite length, we apply a “data window” to signal, which leads to spectral artefacts in frequency domain.

Windowing = Convolution of continuous signal with window function

- Recall: The Fourier transform of a product in time domain is the convolution of the individual Fourier transforms in Frequency domain

- Extracting a segment of a signal in time is the same as multiplying the signal with a rectangular window:



Spectral spreading

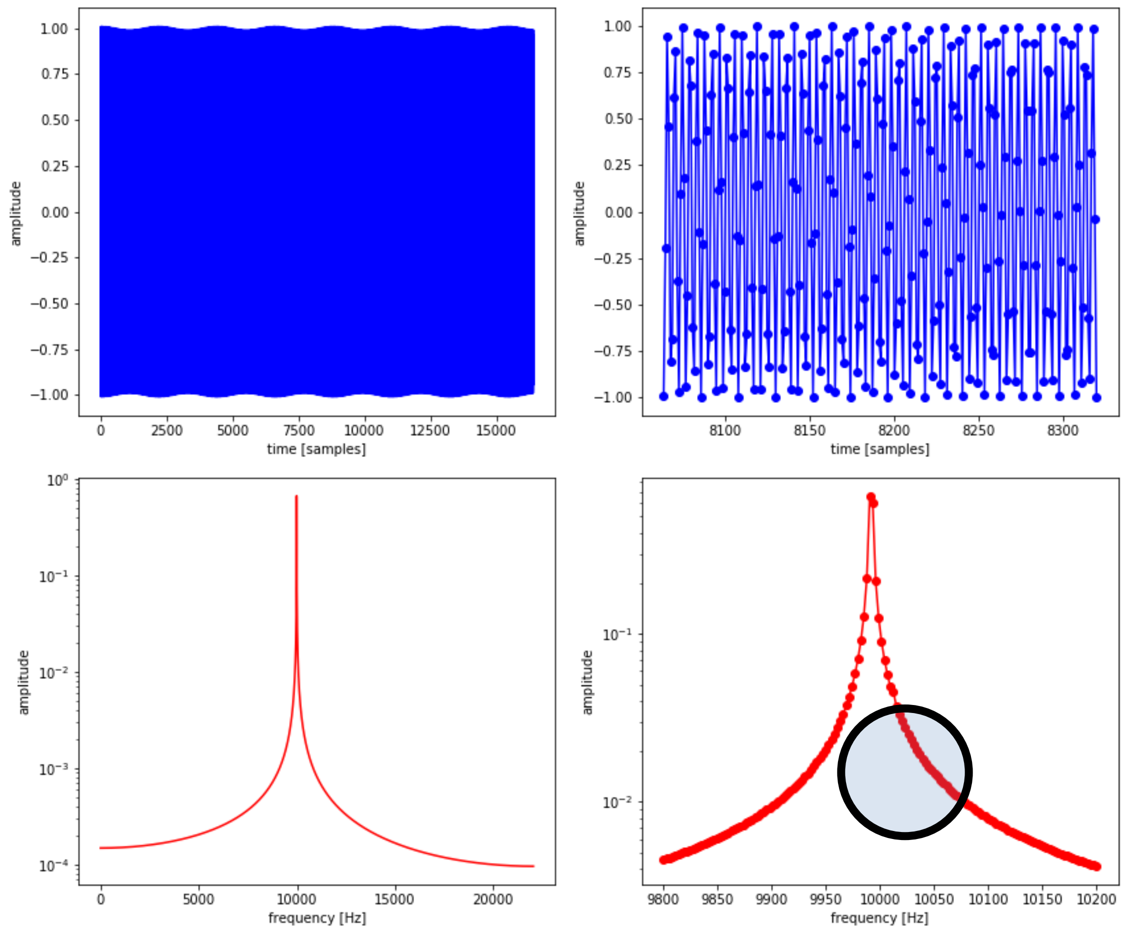
Energy spread out from ω0 to width of 2π/T – reduced spectral resolution.

Leakage

Energy leaks out from the mainlobe to the sidelobes.

Rectangular window example

signal = amp1* sin $(2\pi\ \omega_1 t)$ + amp2 * $\sin(2\pi\ \omega_2 t)$



FFT

ZOOM

amp1 =1
amp2=0.01

$\omega_{1=}$ 2π * 9990 Hz
$\omega_{2=}$ 2π * 10010 Hz

The small signal component is completely masked by the sidelobe of the large signal

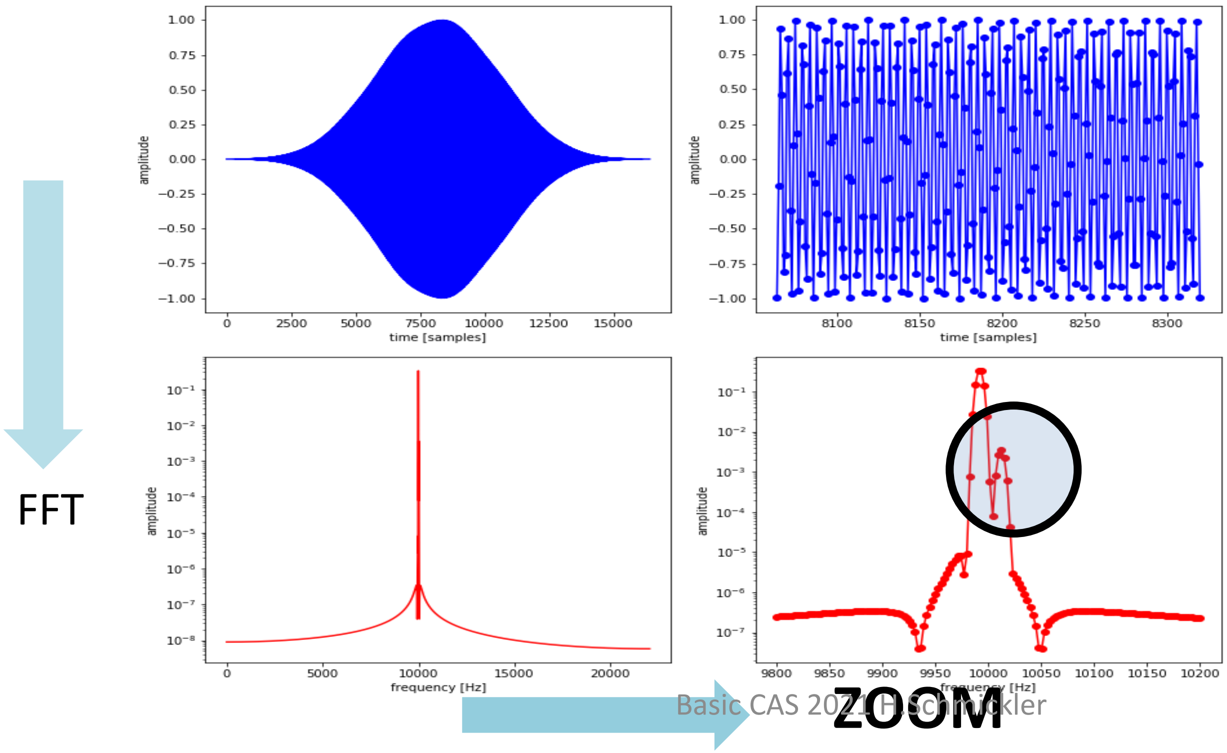
Applying the Blackman-Harris window

Blackman–Harris window

A generalization of the Hamming family, produced by adding more shifted sinc functions, meant to minimize side-lobe levels

$$w[n] = a_0 - a_1 \cos\left(\frac{2\pi n}{N}\right) + a_2 \cos\left(\frac{4\pi n}{N}\right) - a_3 \cos\left(\frac{6\pi n}{N}\right)$$

$$a_0 = 0.35875; \quad a_1 = 0.48829; \quad a_2 = 0.14128; \quad a_3 = 0.01168.$$



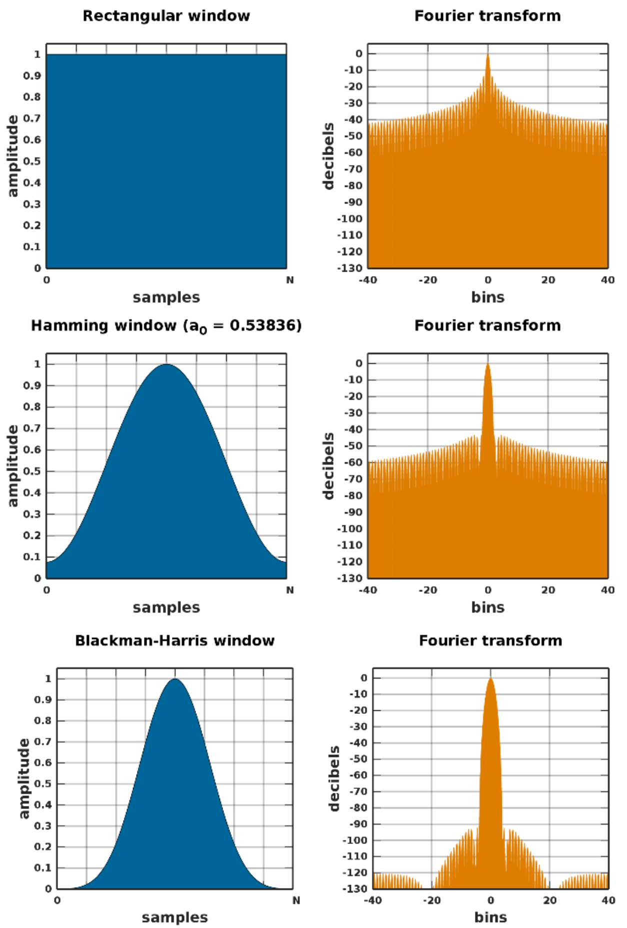
amp1 =1
amp2=0.01

$\omega_1 = 2\pi * 9990$ Hz
$\omega_2 = 2\pi * 10010$ Hz

The small signal component is nicely resolved

Popular window functions

- The following link contains many frequently used window functions, their main features and application:
- https://en.wikipedia.org/wiki/Window_function



The actual choice of the window depends on:

- The signal composition
- The required dynamic range
- The signal to noise ration

remark: every window except the rectangular window is linked to a loss in amplitude (we multiply many samples with almost "zero")
→ reduced S/N up to 6 dB

Improving the frequency resolution of a DFT spectrum

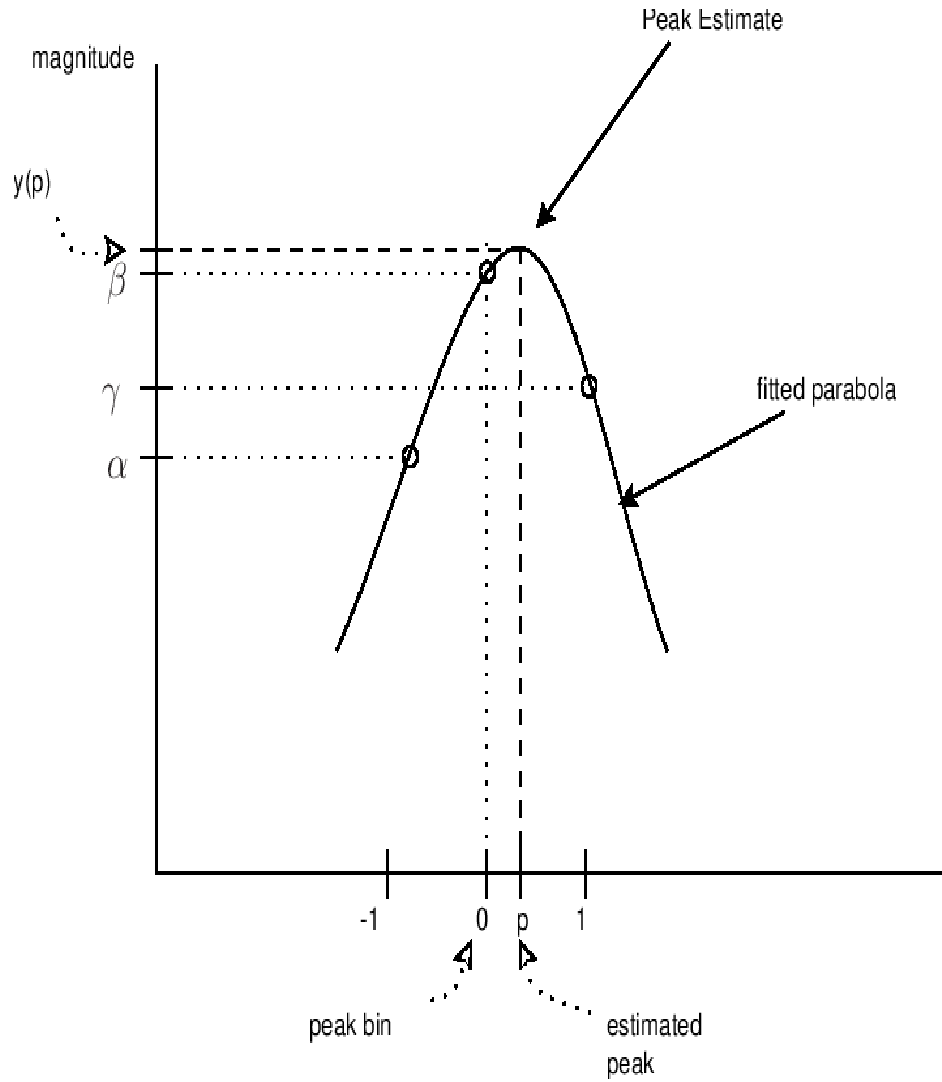
- Recall: basic frequency resolution:
  $\Delta f = 2 * f_{samp}/N_{samp}$

- We can interpolate between the frequency bin with maximum content and the left and right neighbouring bins
- We limit the discussion to "three point interpolation methods"
- The interpolation function is either:

  A) a parabola of the measurements
(:= parabolic interpolation)

  B) a parabola of the log of the measurements
(:= Gaussian interpolation)

- Can get up to $1/N^2$ resolution



Details: https://mgasior.web.cern.ch/mgasior/pap/FFT_resol_note.pdf

Improving the frequency resolution of a DFT spectrum

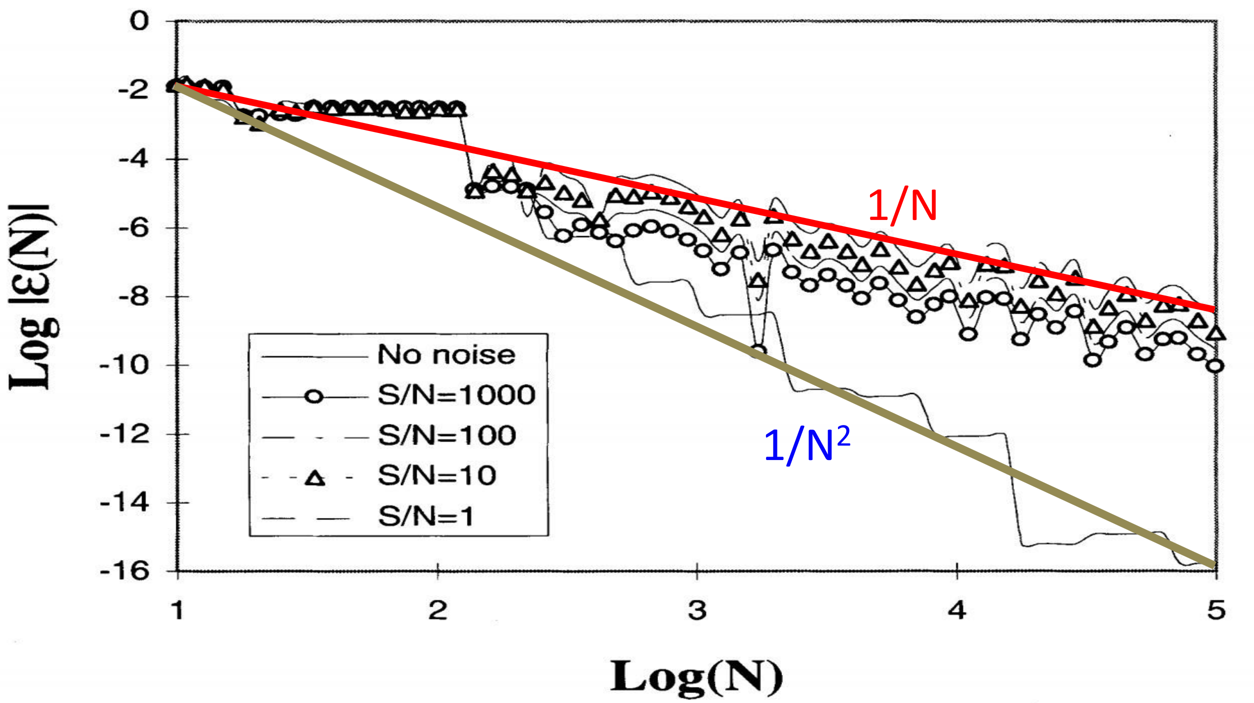
Table 1. Efficiency of the parabolic and Gaussian interpolation with different windowing methods. The windows are characterised by main lobe width, highest sidelobe level and sidelobe asymptotic fall-off. The maximum interpolation error is given as a percentage of the spectrum bin spacing Δ_f. The interpolation gain factor G is defined in (19). Some details concerning the windows and the interpolation errors are given in the Appendix.

Window	Main lobe width [bin]	Highest sidelobe [dB]	Sidelobe asymptotic fall-off [dB/oct]	**Parabolic interpolation**		**Gaussian interpolation**	
				Error max. [% of Δ_f]	**Gain factor G**	**Error max. [% of Δ_f]**	**Gain factor G**
Rectangular	2	-13.3	6	23.4	2.14	16.7	2.99
Triangular	4	-26.5	12	6.92	7.23	2.08	24.1
Hann	4	-31.5	18	5.28	9.47	1.60	31.2
Hamming	4	-44.0	6	6.80	7.35	1.60	31.2
Blackman	6	-68.2	6	4.66	10.7	0.578	86.5
Blackman-Harris	6.54	-74.4	6	4.18	12.0	0.476	105
Nuttall	8	-98.2	6	3.51	14.2	0.314	159
Blackman-Harris-Nuttall	8	-93.3	18	3.34	15.0	0.314	159
Gaussian $L = 6\ \sigma$	6.96	-57.2	6	4.95	10.1	0.240	208
Gaussian $L = 7\ \sigma$	10.46	-71.0	6	3.80	13.2	0.0516	970
Gaussian $L = 8\ \sigma$	11.41	-87.6	6	2.95	17.0	0.00869	5756

$$Gain factor G := \frac{\Delta f}{2x Error max.}$$

from: https://mgasior.web.cern.ch/mgasior/pap/FFT_resol_note.pdf

A little summary on frequency resolution



Taken from: R. Bartolini et al, Precise Measurement of the Betatron tune, Proceedings of PAC 1995, Vol. 55, pp 247-256

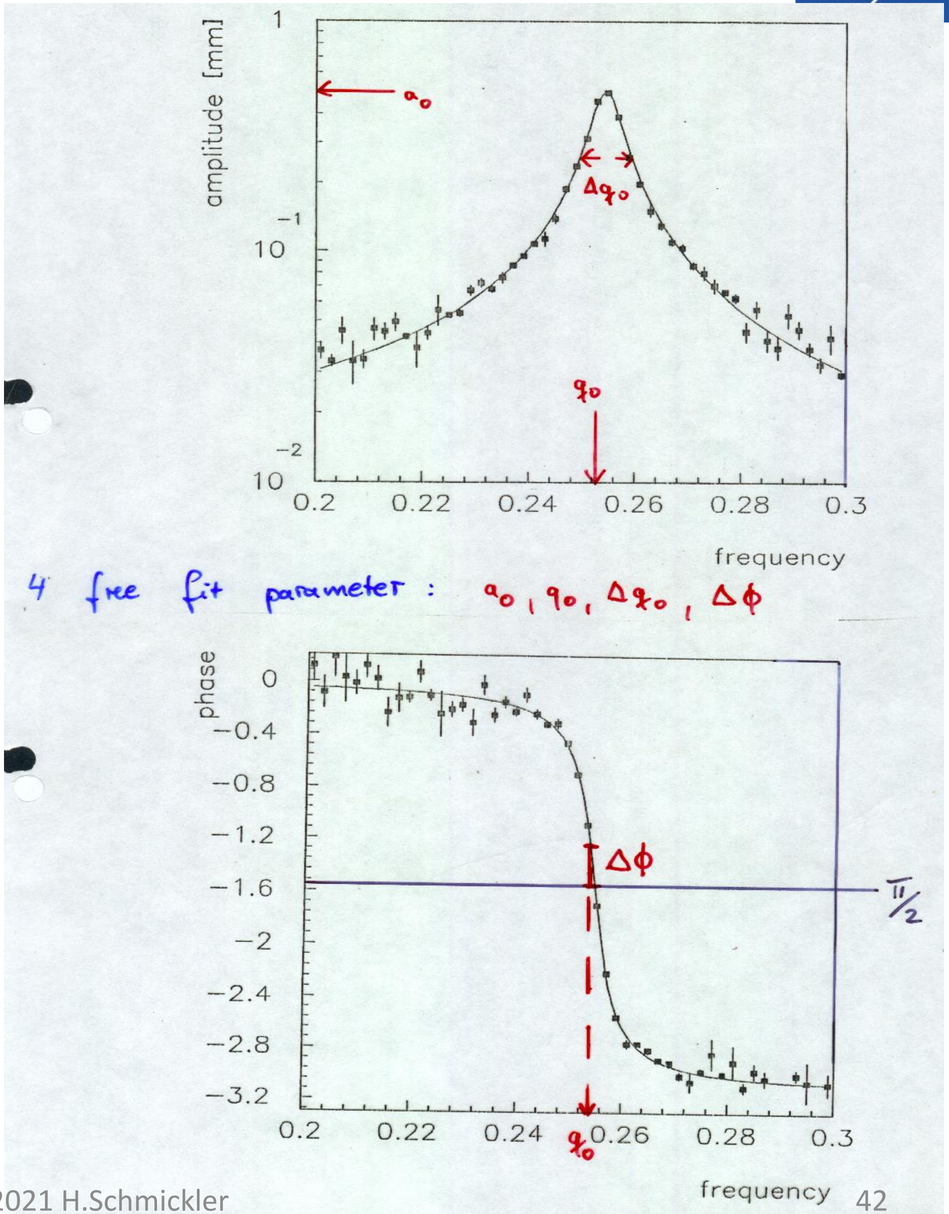
- Frequency measurement error ε(N) as function of log (N) for different S/N ratios
- Basic FFT resolution proportional to 1/N
- Plot shows result for interpolation using Hanning window.
- With interpolation and no noise proportional to $1/N^2$

Other method: Network analysis

1. Excite beams with a sinusoidal carrier

1. Measure beam response

1. Sweep excitation frequency slowly through beam response



Analysis of non-stationary spectra

- Stationary Signal
  - Signals with frequency content unchanged in time
  - All frequency components exist at all times

    → ideal situation for Fourier transform (FT)
    (orthonormal base functions of Fourier transform are infinitely long, no time information when spectral component happens)

- Non-stationary Signal
  - Frequency composition changes in time

    → need different analysis tools

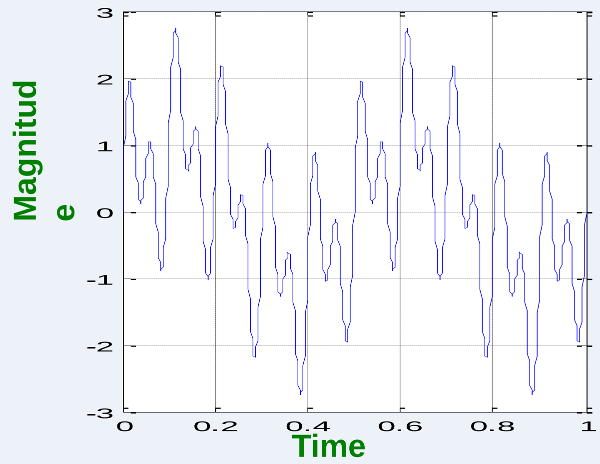
  - One example: the "Chirp Signal"

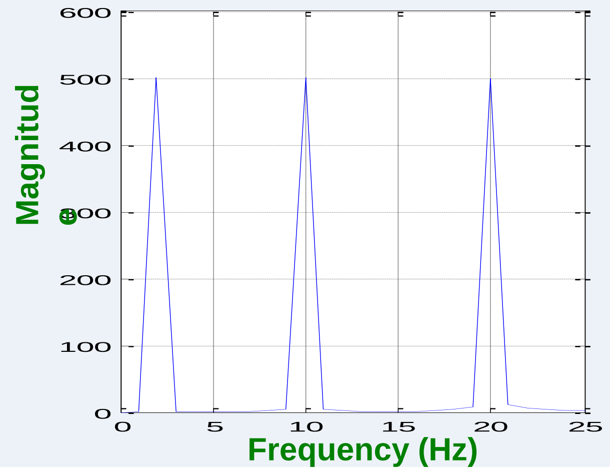
The CERN Accelerator School

Example of simple stationary or non-stationary signals

2 Hz + 10 Hz + 20Hz

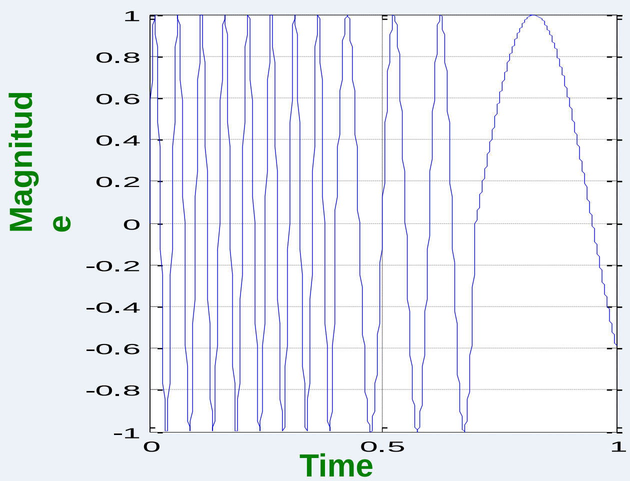
Stationary

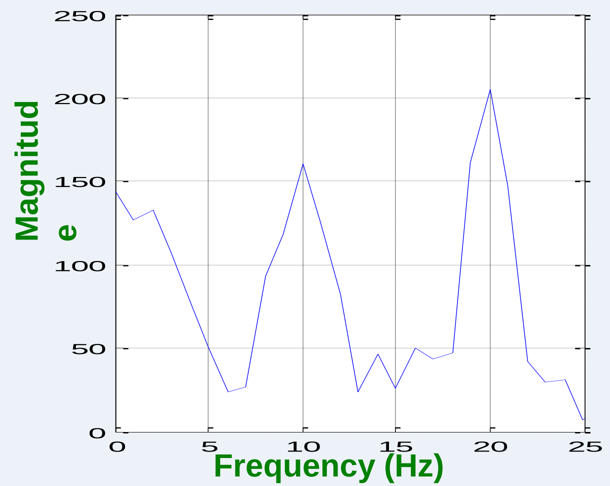




0.0-0.4: 20 Hz +
0.4-0.7: 10 Hz +
0.7-1.0: 2 Hz

Non-Stationary



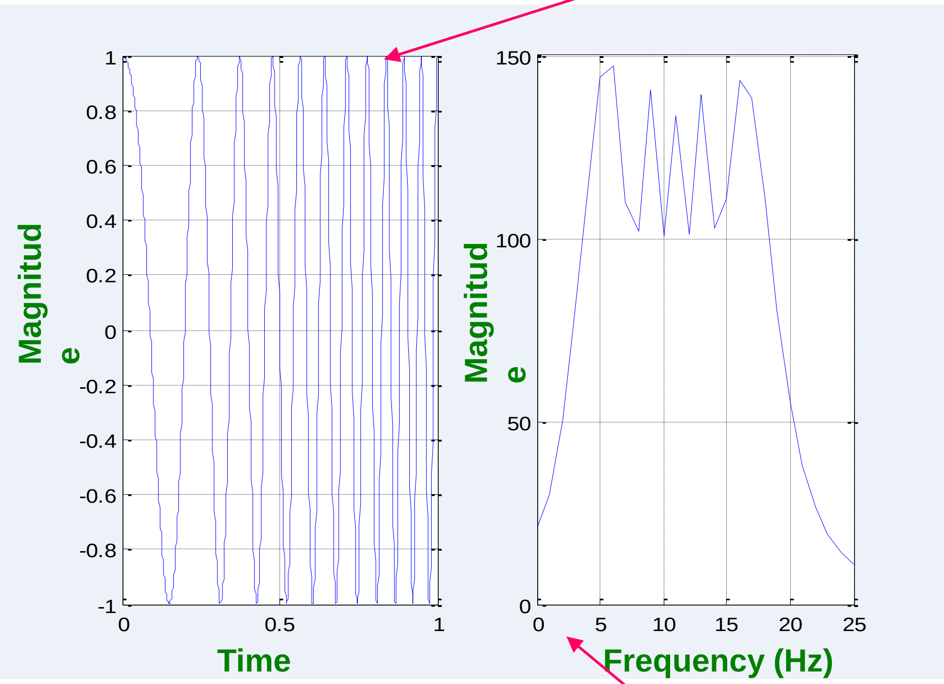


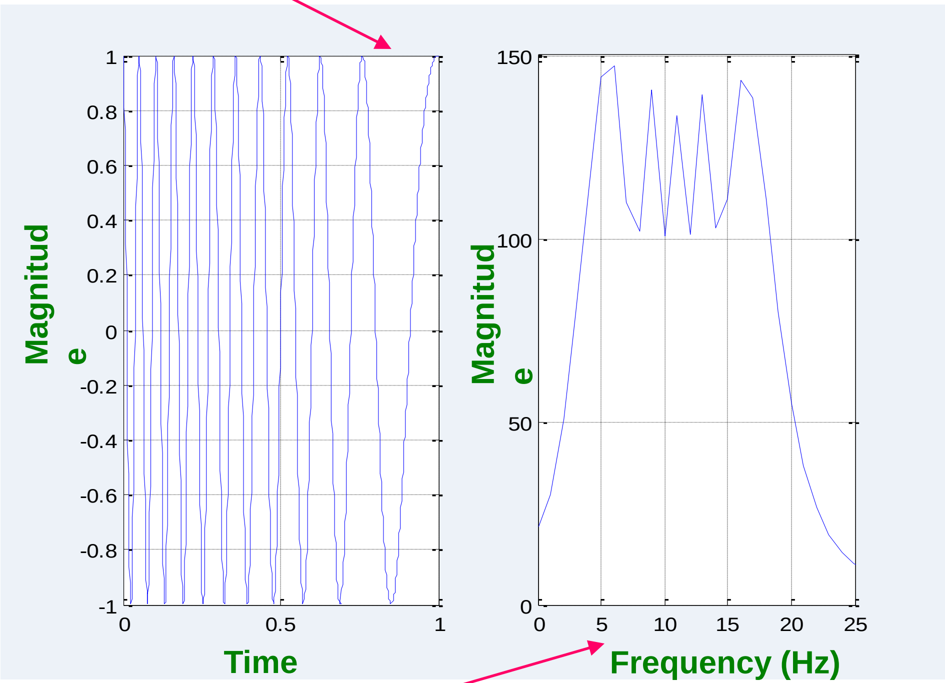
Upward or downward chirp

linear chirp: 2 Hz to 20 Hz

linear chirp: 20 Hz to 2 Hz

Different in Time Domain



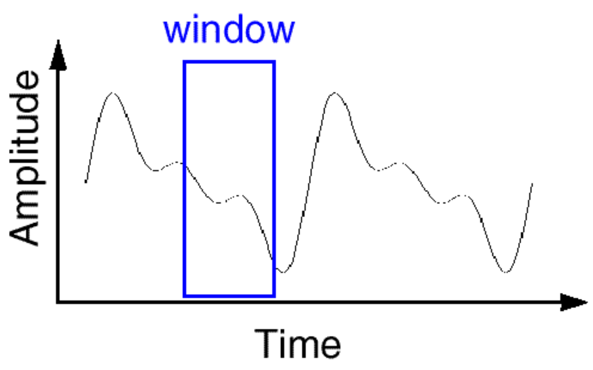


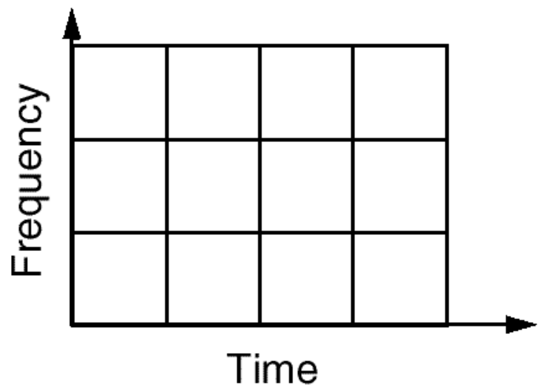
Same in Frequency Domain

At what time a frequency component occurs? FT can not tell!

Short Time Fourier Analysis

- In order to analyze small section of a signal, Denis Gabor (1946), developed a technique, based on the FT and using *windowing* :
  Short Time Fourier Transform:= STFT

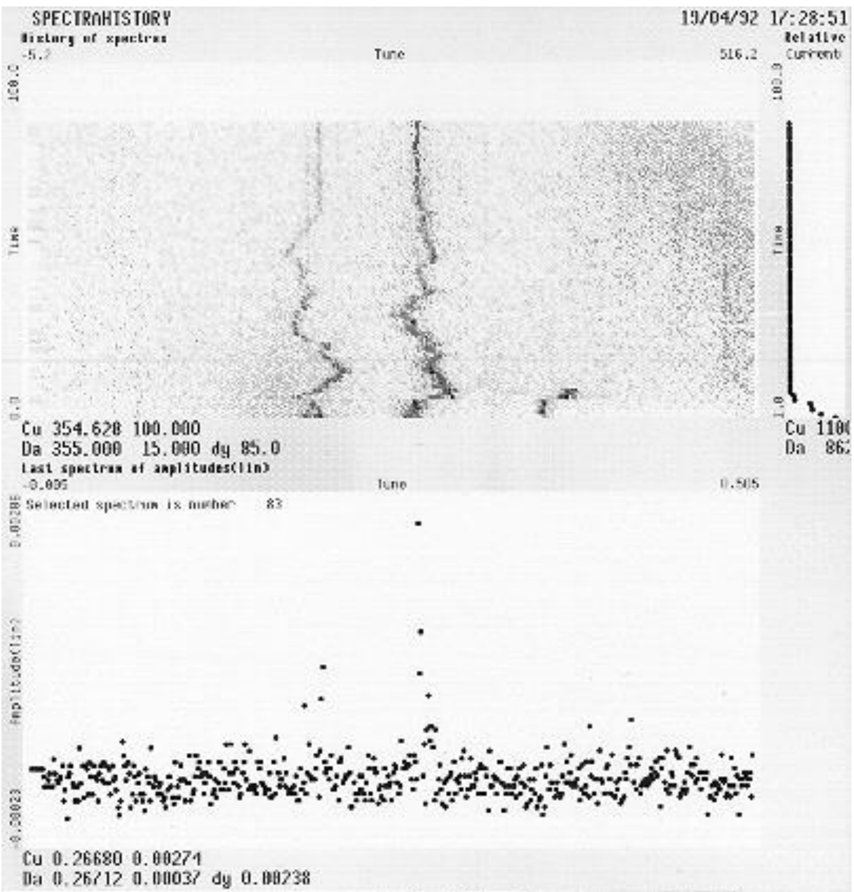


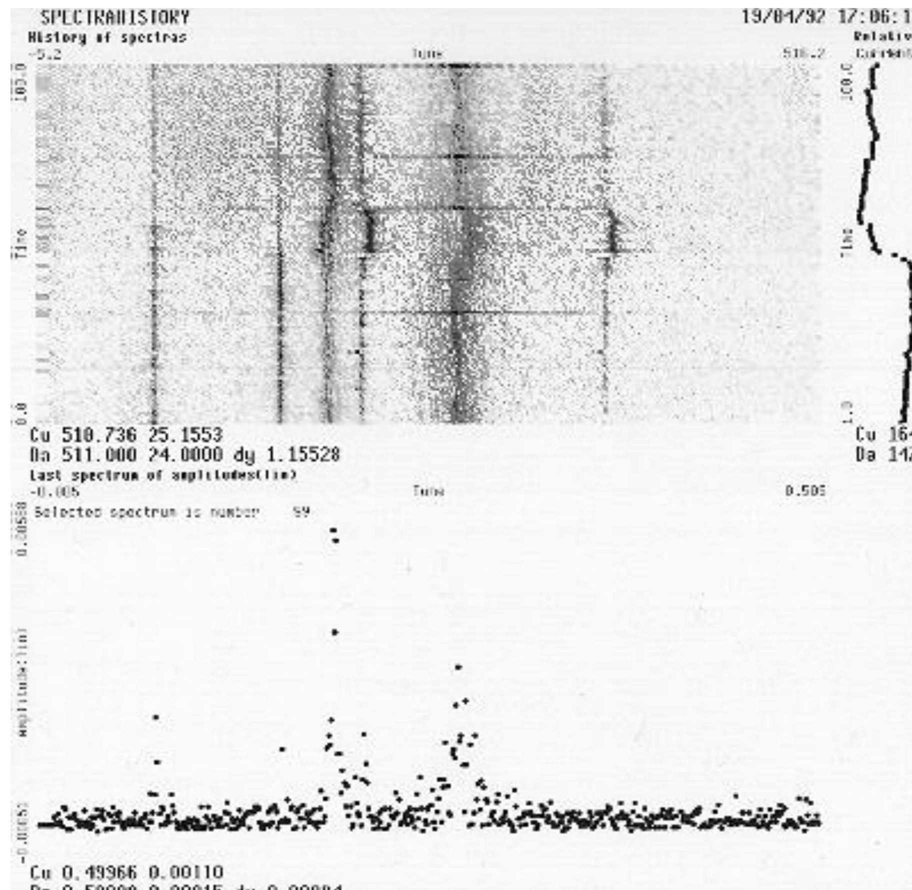


- A compromise between time-based and frequency-based views of a signal.
- both time and frequency are represented in limited precision.
- The precision is determined by the size of the window.
- Once you choose a particular size for the time window - it will be the same for all frequencies.

Time Resolved Tune Measurements

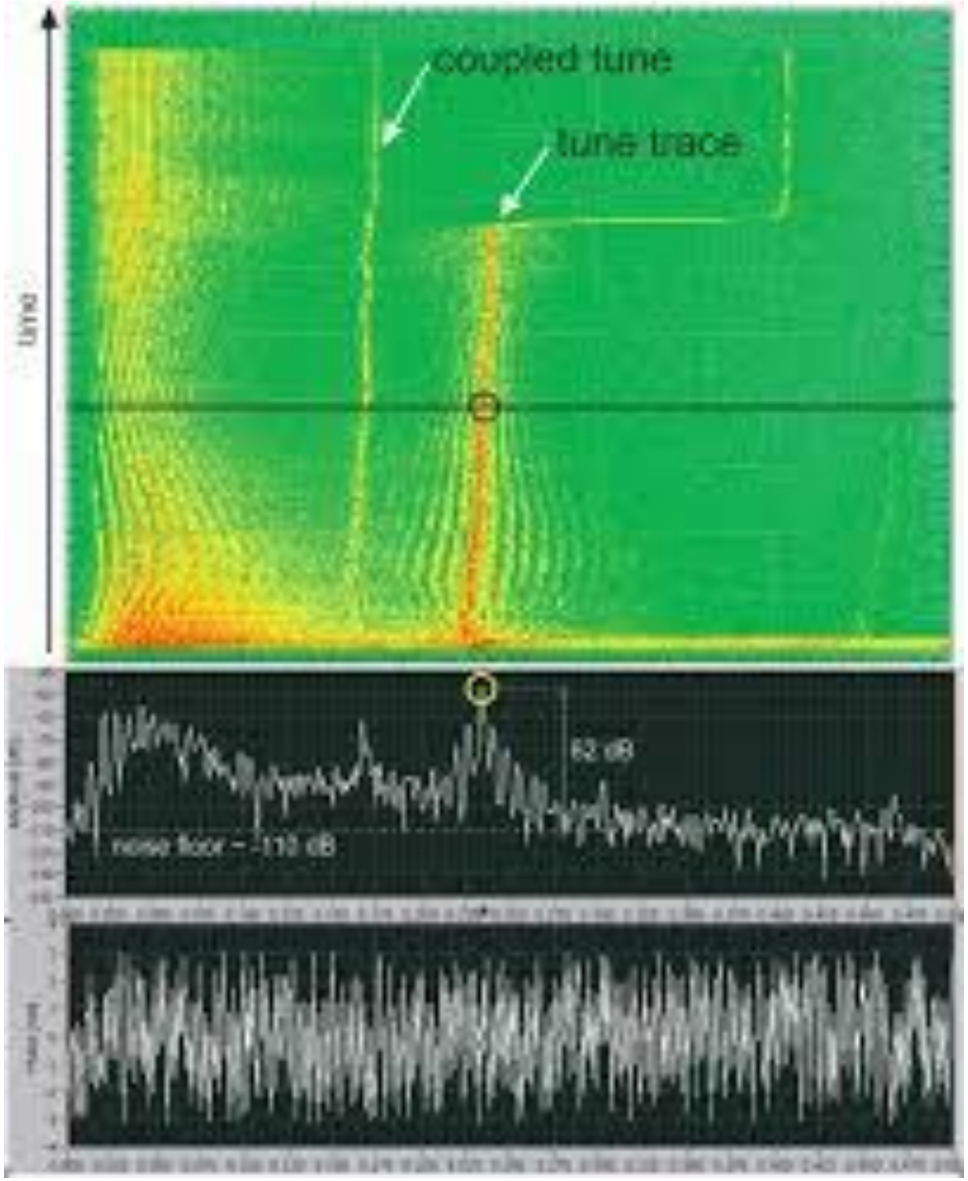
- To follow betatron tunes during machine transitions we need time resolved measurements. Simplest example:
  - repeated FFT spectra as before (spectrograms)





STFT display: Spectogram

- A very useful form of displaying the result of a STFT is a spectrogram, i.e a 3D view of many consecutive Fourier transforms, which “slide” along the time series of data.



Summary

- Single beam passage in a detector produces a signal with a continuous frequency spectrum. The shorter the bunch, the higher the frequency content.
- Repetitive bunch passages produce a line spectrum. They are called revolution harmonics.
  Details of the bunch pattern, differences in bunch intensities etc. determine the final spectral distribution.
- Transverse or longitudinal oscillations of the bunch around the equilibrium produce sidebands around all revolution harmonics.
- These sidebands are used for the measurement of the betatron tunes or the synchrotron tune.

- The standard tool for obtaining spectral information is a Fourier transform (FFT) of the time sampled signals.
- Windowing and interpolation allow higher resolution measurements.
- Spectograms or STFTs are consecutive FFTs of larger datasets, which allow to follow time varying spectra.

Appendix I: Python Code for bunch pattern display

Appendix Ia: Python code for bunch pattern simulation 1st part

```
import numpy as np
from numpy import fft
import matplotlib.pyplot as plt

N=16384

NBUNCH=100
sigmax = 0.5


deltax=10

T=1/N

NLEFT=-50
NRIGHT=50

x1= np.linspace(NLEFT,N-NLEFT,N)
xtime=np.linspace(NLEFT,NBUNCH*deltax + NRIGHT,N)
IB=0
y=NBUNCH*np.exp(-(x1*x1)/(2*sigmax*sigmax))
ytime=NBUNCH*np.exp(-(xtime*xtime)/(2*sigmax*sigmax))
y1=0
y2=0
y3=0
ytime=0

while True:

  y1=y1+np.exp(-(x1-IB*deltax)*(x1-IB*deltax)/(2*sigmax*sigmax))
  ytime=ytime+np.exp(-(xtime-IB*deltax)*(xtime-IB*deltax)/(2*sigmax*sigmax))
  IB=IB+1
  if IB==NBUNCH:
    break
```

Appendix Ib: Python code for bunch pattern simulation 2nd part

```
fftу=(fft.fft(y))
ffty1=(fft.fft(y1))
x2=np.linspace(0.0,500,N/2)
y2=2.0*np.abs(ffty1[:N//2])/float(N)
y3=2.0*np.abs(ffty[:N//2])/float(N)

plt.rcParams["figure.figsize"] = [15,4]
plt.subplot(1,2,1)

plt.plot(xtime,ytime,'b-')

plt.ylabel('amplitude')
plt.xlabel('time [nsec]')

plt.subplot (1,2,2)

plt.plot (x2,y3,'r-')
plt.plot (x2,y2,'b-')
plt.ylabel('amplitude')
plt.xlabel('frequency [MHz]')

plt.tight_layout()
plt.savefig (‘whatever.png')

plt.show()
```