Real Time Control of Beam Parameters

Micha Dehler

Micha.Dehler@psi.ch

Paul Scherrer Institut, CH-5232 Villigen PSI

Abstract

Real time feedback systems always are a trade off between bandwidth, latency and noise. System bandwidth is often not only dictated by the time structure of the beam signal, but also by latency considerations. The thermal noise floor relating bandwidth and noise power is only one design criterion.

High bandwidth, low latency systems often have to compromise on the resolution of the analog to digital and digital to analog conversion as well as digital data width giving rise to discretisation noise in amplitude and time. An efficient system design needs an explicit model specifying the different types and locations of the noise sources.

The lecture demonstrates the approach for existing systems, an orbit feedback and a bunch by bunch stabilization system, discusses choices for hardware and software and takes a look at how future technology trends will affect system design and layout.

Motivation

- Characteristics of real time system (what we understand as a real time system...)
- Hardware challenges
- Data flux and format
- Controller design

Outline

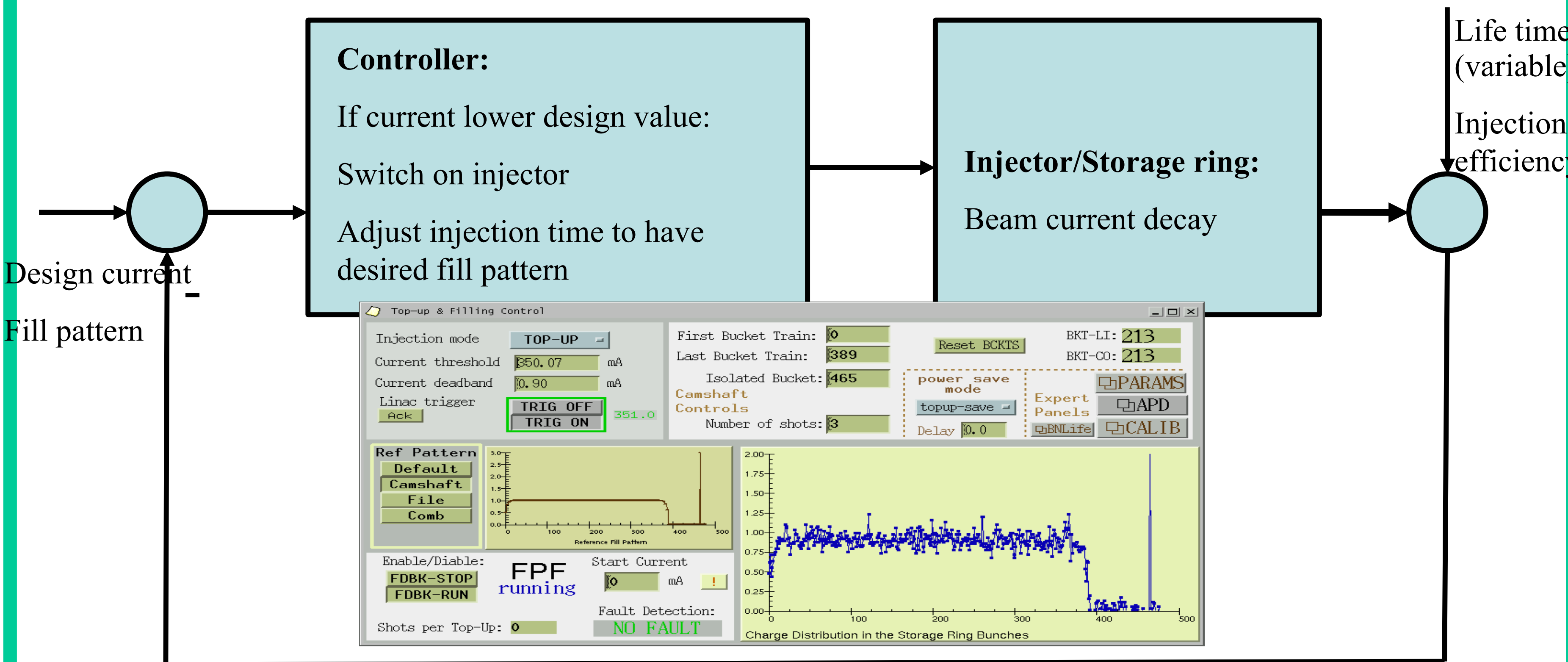
- What is a RT system?
- Global orbit feedback
- Bunch by bunch feedback system
- Recapitulation and Outlook

Why we are not really talking about RT systems?

What wikipedia says about RT systems:

- A system is said to be **real-time** if the correctness of an operation depends not only upon the logical correctness of the operation but also upon the time at which it is performed. The classical conception is that in a **hard** or **immediate real-time system**, the completion of an operation after its deadline is considered useless - ultimately, this may lead to a critical failure of the complete system. A **soft real-time system** on the other hand will tolerate such lateness, and may respond with decreased service quality (e.g., dropping frames while displaying a video).
- Hard real-time systems are used when it is imperative that an event is reacted to within a strict deadline..........
- Soft real-time systems are typically those used where there is some issue of concurrent access and the need to keep a number of connected systems up to date with changing situations.....
- It is important to note that hard versus soft real-time does not necessarily relate to the length of time available.....

A fill pattern feedback – a RT system



- Fill pattern has influence on coupled bunch modes in SLS
- Measurements within a second, injection every minute
- Certainly a (soft?) RT system according to wiki definition!!
- Do you want to discuss this type of system?

Our definition of a RT system

- Effective bandwidth of the system is dominated by latencies and delays
- Noise sources in the system have inherent bandwidths equal to or even exceeding this effective bandwidth.

Definition of a RT system (in the vein of Murphy‘s law):

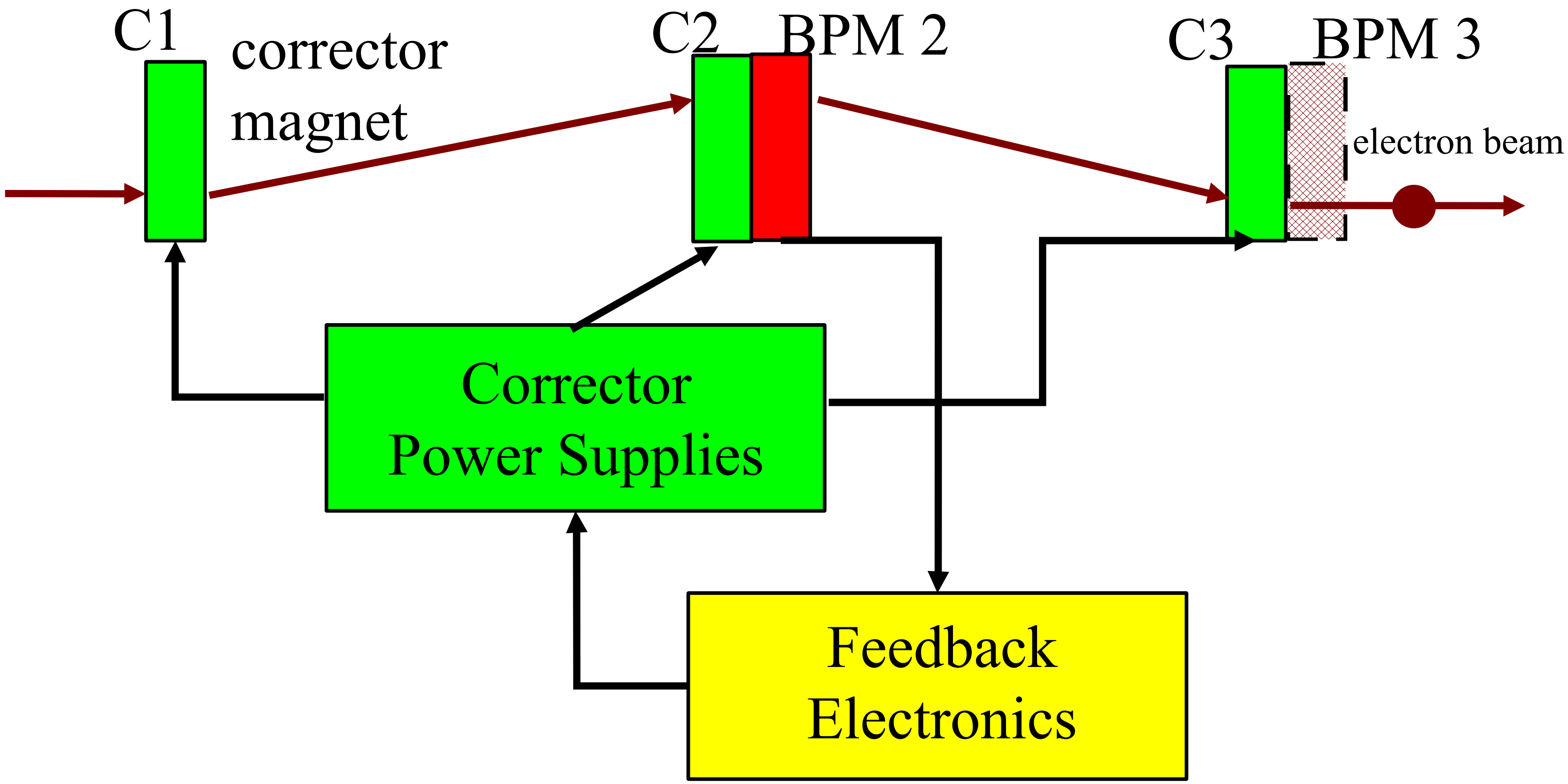
A system is real time, if and only if its response is too late anyway to correct the situation, it is supposed to control.

Typical RT systems

- Orbit stabilisation
  - At RF frequencies
    - Bunch by bunch feedbacks (storage rings)
    - Intra bunch train feedbacks (linear machines)
  - Orbit feedbacks
    - Local
    - Global
    - Staggered
- Tune feedbacks (+ other optics)
- Subsystems
  - RF amplitude and phase loops
  - Magnet power supplies

Orbit feedbacks

Local feedback



- Control variable: Bump strength = set of values for three correctors C1...C3
- Bump over four controllers (=Offsets in BPM 2 and 3) also controls angles
- Within first order accuracy sets only orbit offset at associated BPM, no other influences
- Reality: Changing bump changes orbit path lengths

Global feedbacks

- Using n correctors c_i creates m BPM offsets x_i (MIMO feedback problem)

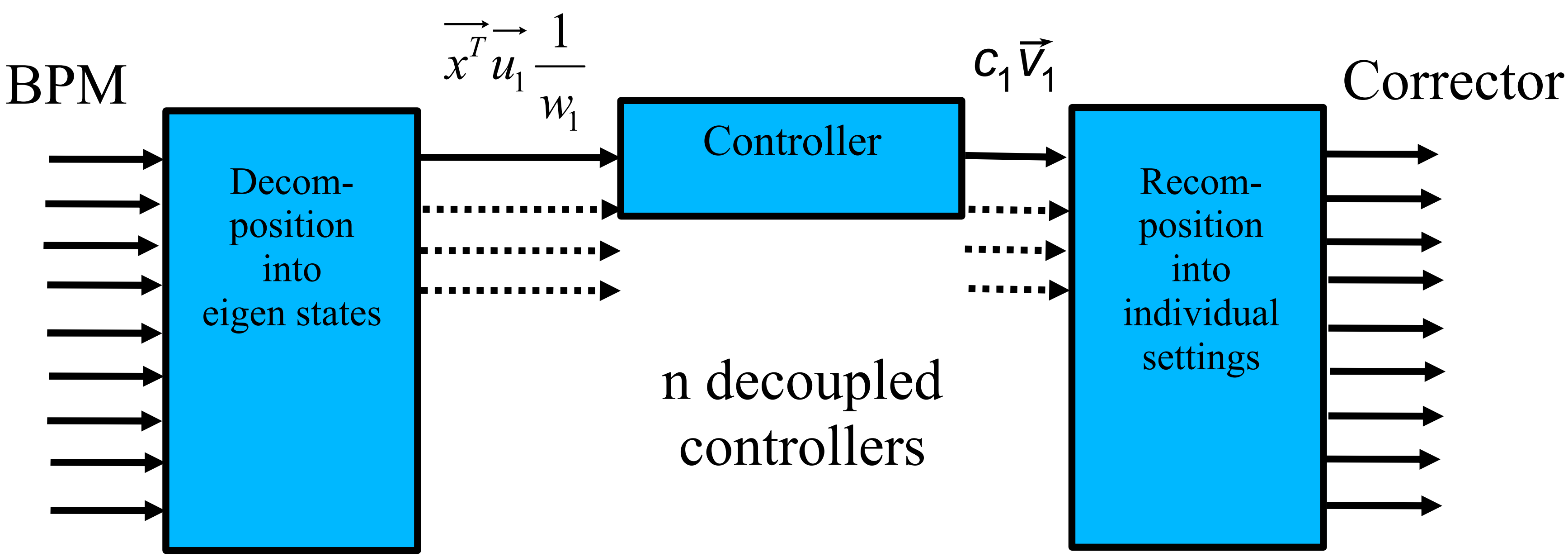
$$\vec{x} = M \vec{c}$$

- Singular value decomposition (SVD):

$$M = \sum_{k=1}^{n} \vec{u_k}\, w_k\, \vec{v_k^T}$$

- create pseudoinverse of M:

$$\tilde{M}^{-1} = \sum_{k=1}^{n} \vec{v_k} \frac{1}{w_k} \vec{u_k^T}$$



Dynamic behaviour

The most general (linear) case:

$$M(j\omega) = \sum_{k=1}^{n} \overrightarrow{u_k}(j\omega) w_k(j\omega) \overrightarrow{v_{k^T}}(j\omega)$$

all parts of the decomposition are frequency dependent (Good luck! Next time, design a better system!)

The RH and LH eigenvectors are independent of frequency, need different controller characteristics for different eigenvalues:

$$M(j\omega) = \sum_{k=1}^{n} \overrightarrow{u_k} w_k(j\omega) \overrightarrow{v_{k^T}}$$

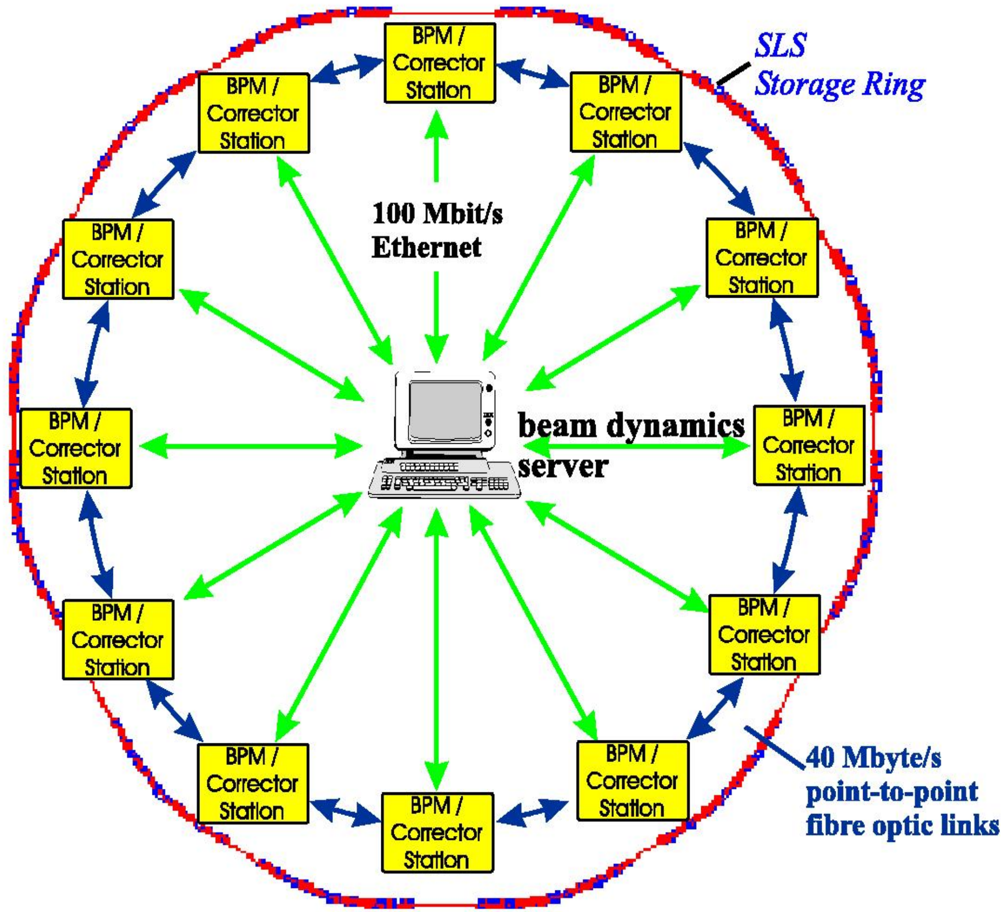
Have scalar frequency dependency, SVD is static, controllers differ only in the amplification factor:

$$M(j\omega) = f(j\omega) \sum_{k=1}^{n} \overrightarrow{u_k} w_k \overrightarrow{v_{k^T}}$$

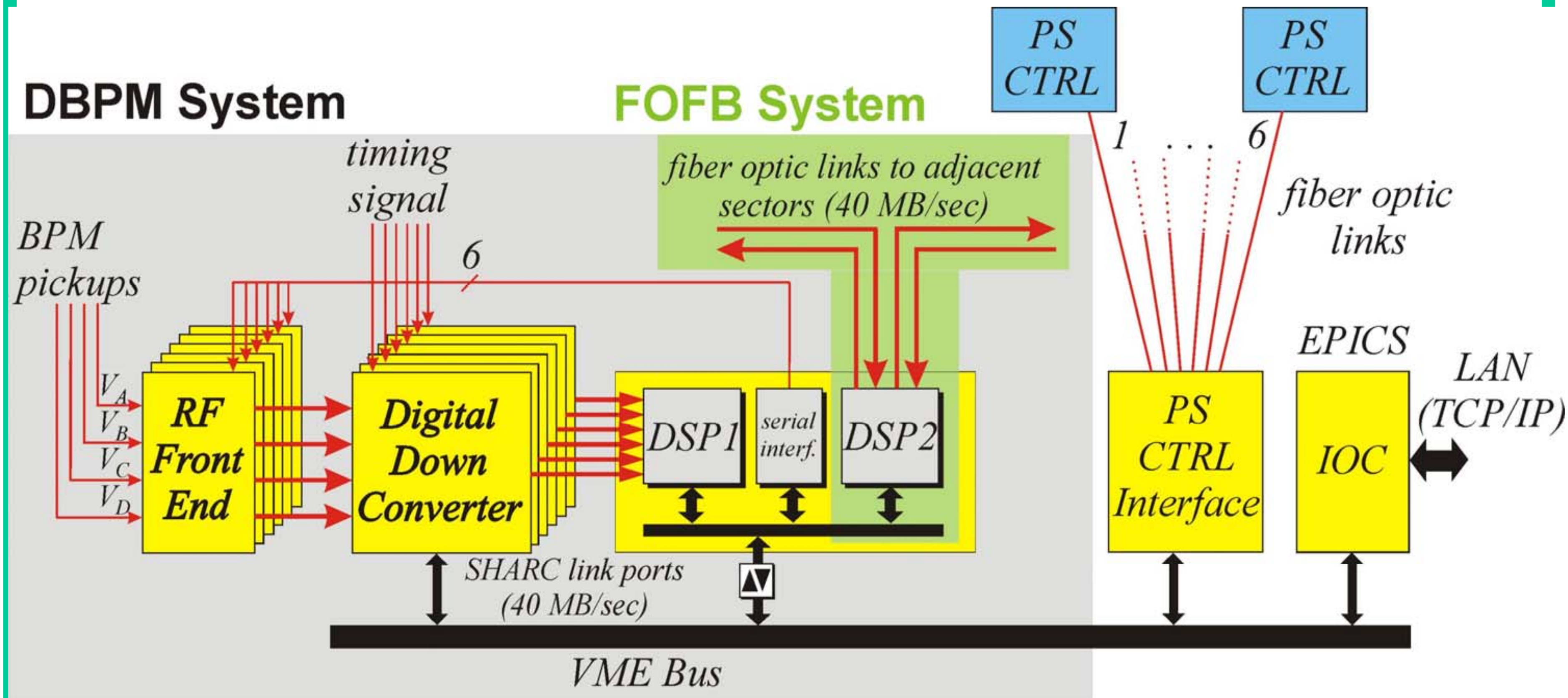
difficult to obtain?

Typical layout for a global orbit feedback (SLS)

•Each 6 BPM and 6 correctors per BPM/Corrector station
•Fast communication links between station for quick corrector computation
•Central server for SVD computation, offline diagnostics and operator interface and coupling to RF systems

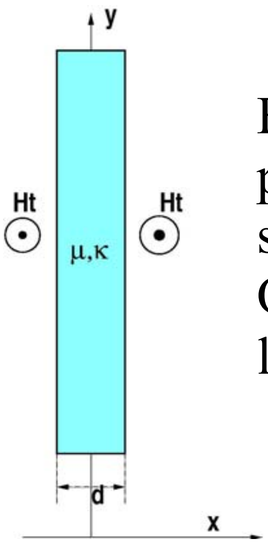


A BPM/Corrector station

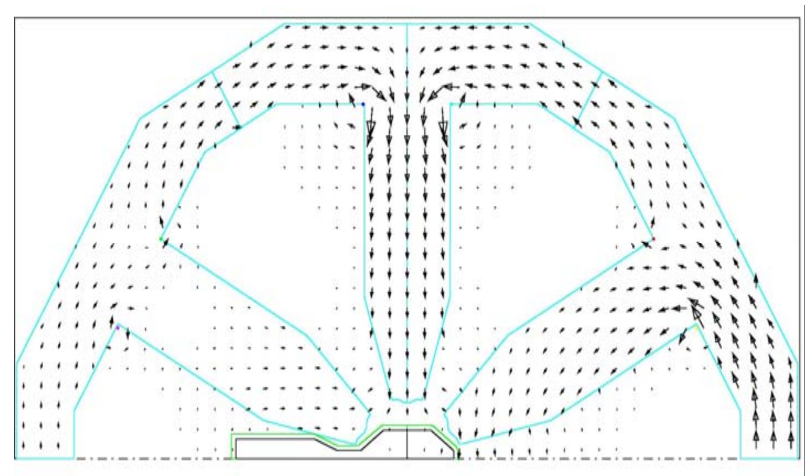


- Total system latency 1.5 ms at 4 kS/sec. sample/correction rate
- Bandwidth RF front end/Digital Down Converter: 2 kHz
- Power supply bandwidth: 2 kHz (... a feedback on his own ...)
- Total system bandwidth limited by other components

Corrector band width due to core losses



Field excited by coils has to propagate into lamination sheets making up corrector
Only essential parameter lamination thickness

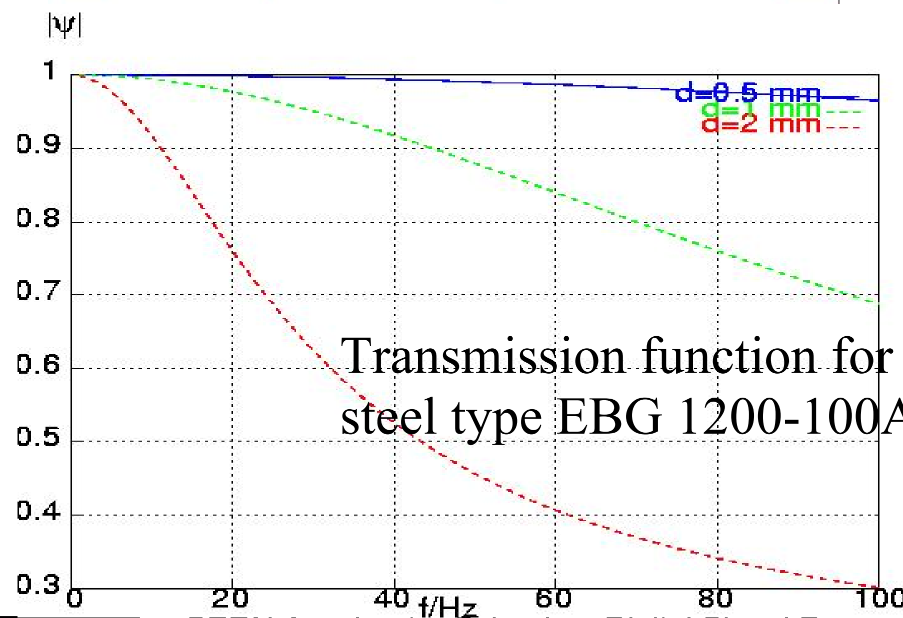


Frequency dependencies

$$\frac{\Psi(j\omega)}{\Psi_{DC}} = \frac{2}{kd}\tan\left(\frac{kd}{2}\right)$$

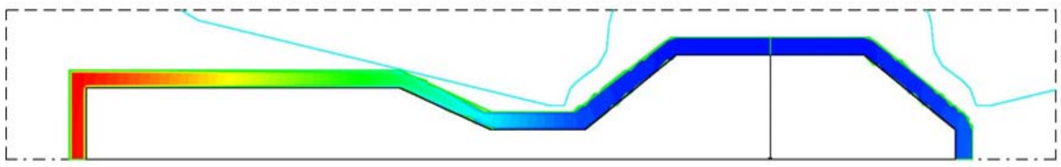
$$k^2 = -j\omega\mu\kappa$$

with μ and κ the permeability and conductivity of the core and d the lamination thickness



Transmission function for steel type EBG 1200-100A

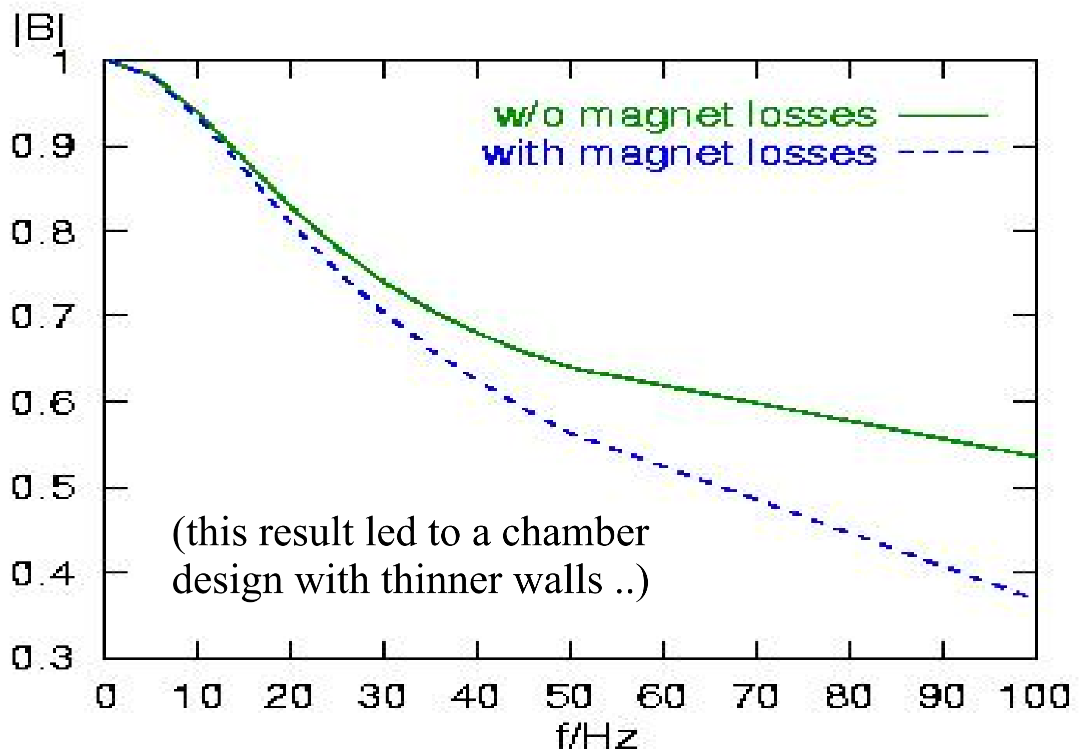
Magnetic Fields have to propagate into the vacuum chamber



Symmetric half of vacuum chamber

Loss power density in vacuum chamber wall due to eddy currents

- Frequency dependency of losses lead to attenuations and phase shifts of field seen by beam (plus even spurious quadrupole momenta)
- Result via numerical computations
- **This and the core losses are the main determinants of feedback speed!**



Design objectives for the controller

- Minimize overshoot, settling time for changing control parameters (E.g. Feedbacks for ramping operations)
- Change location of instable poles
- Avoid saturation in selected parts (ADC, DAC, amplifiers, magnet power supplies)
- Minimize fluctuations due to internal/external noise

Excitation function – the ground motion

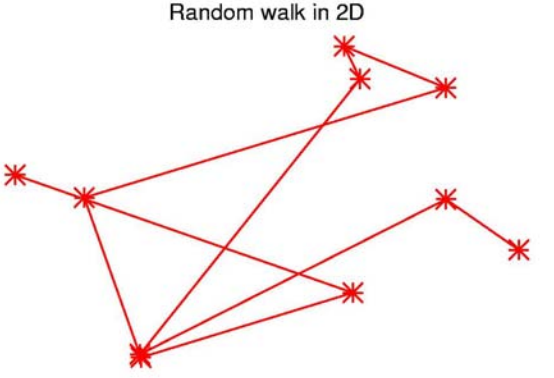
- Fundamental drifts following random walk/Brownian motion
  - ATL law describing rms drift of points at distance L over time T:

$$E\left(\delta x^2\right) = ATL;\ \ A \approx 10^{-5}\,\mu m^2/(sm)$$

  - Corresponding power density:

$$S(j\omega) = \frac{2A}{\omega^2}$$

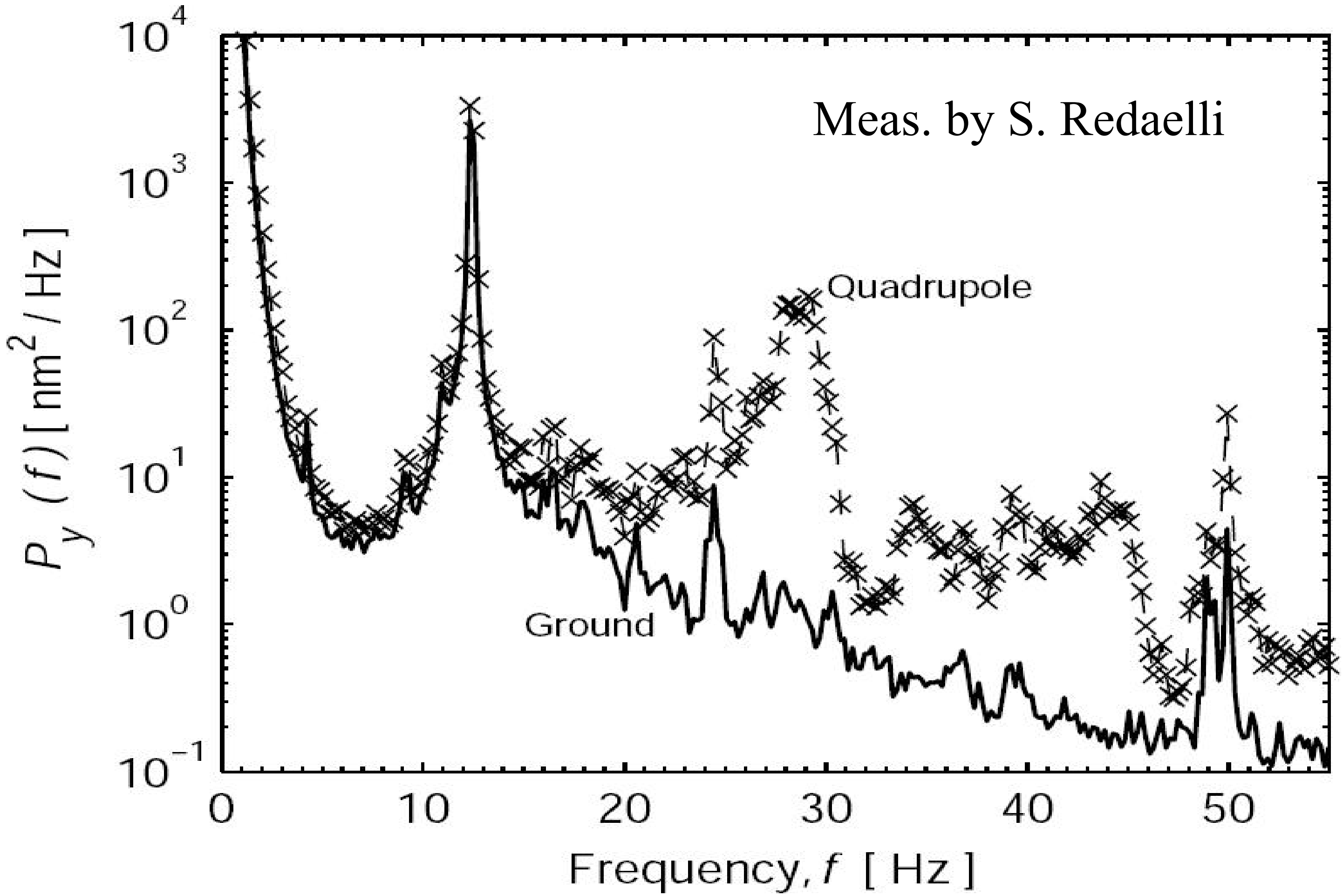
(only valid for wavelengths smaller than the



Other external ground vibrations due to e.g. human traffic

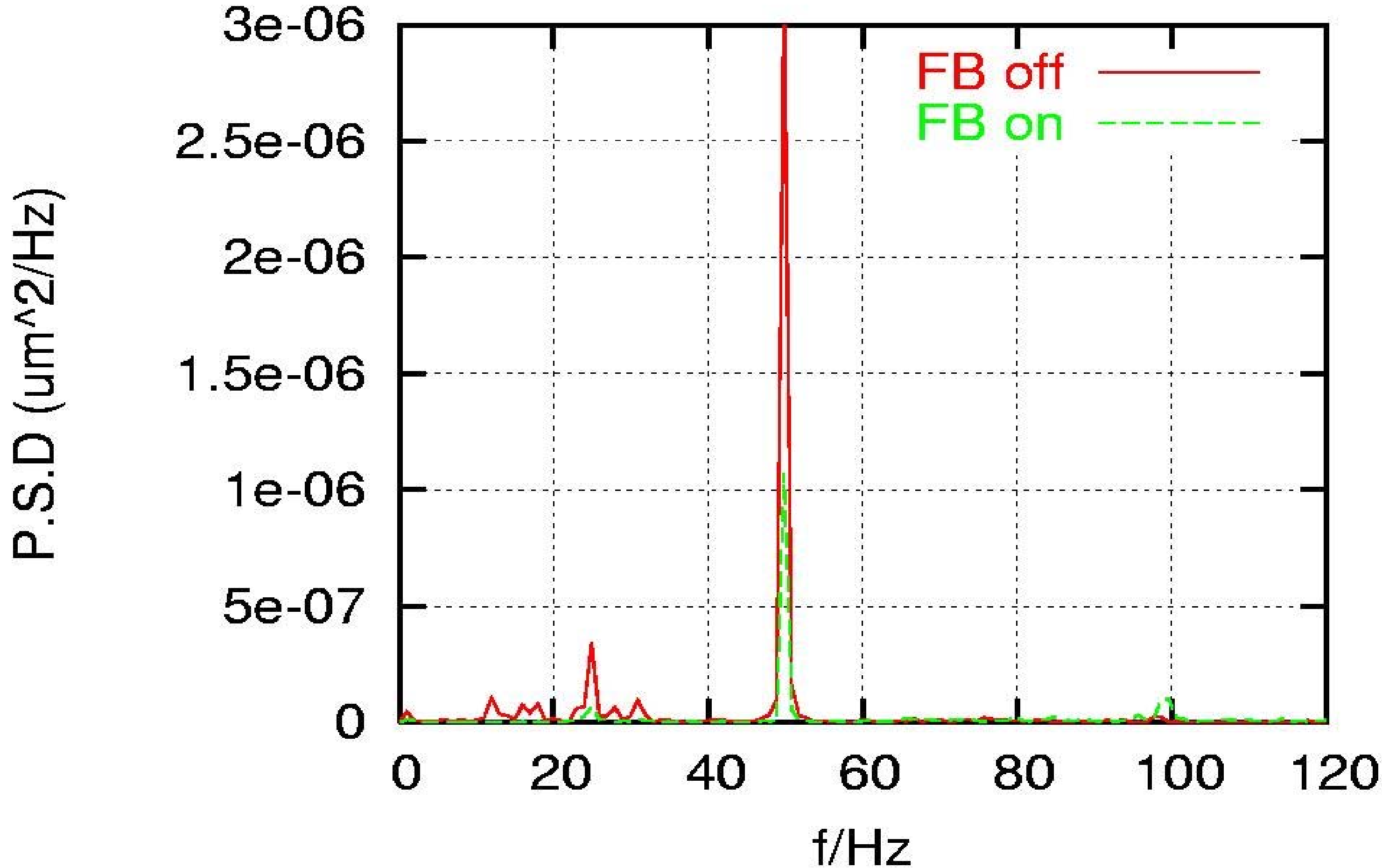
Systematic excitations (e.g. Compressor noise, 50/60 Hz mains)

Ground spectrum gets transformed by girder transfer function

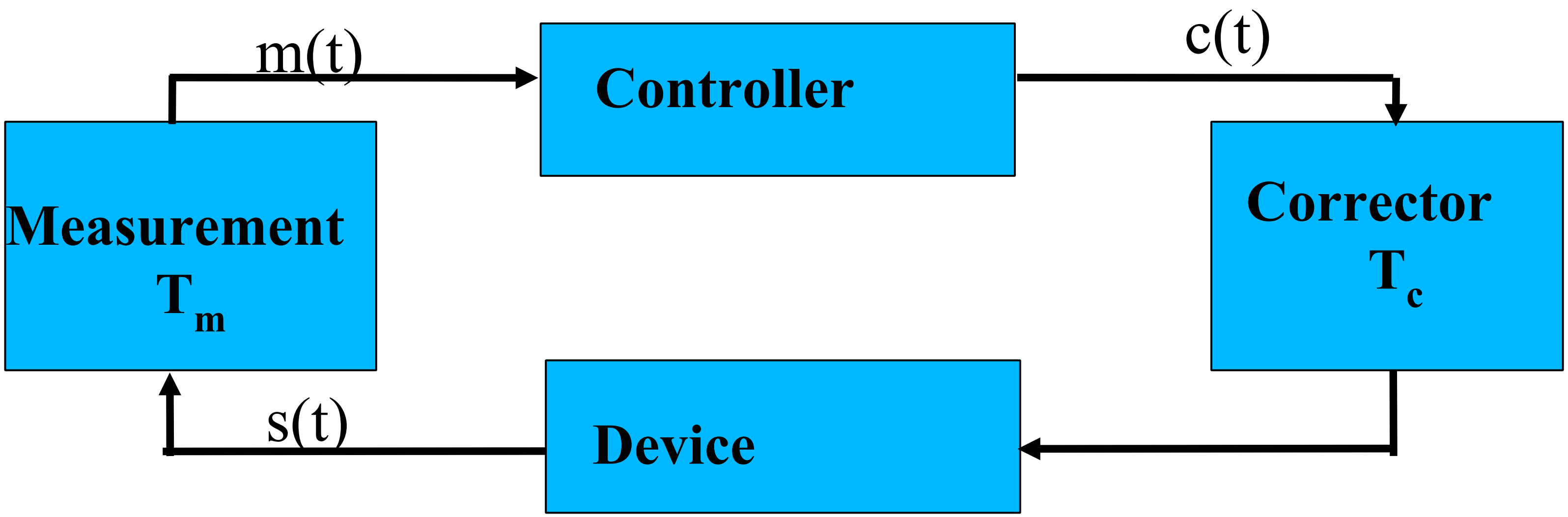


Side remark: Best put accelerator directly on the ground (see e.g. SCSS accelerator layout ..)

Performance using a PID controller



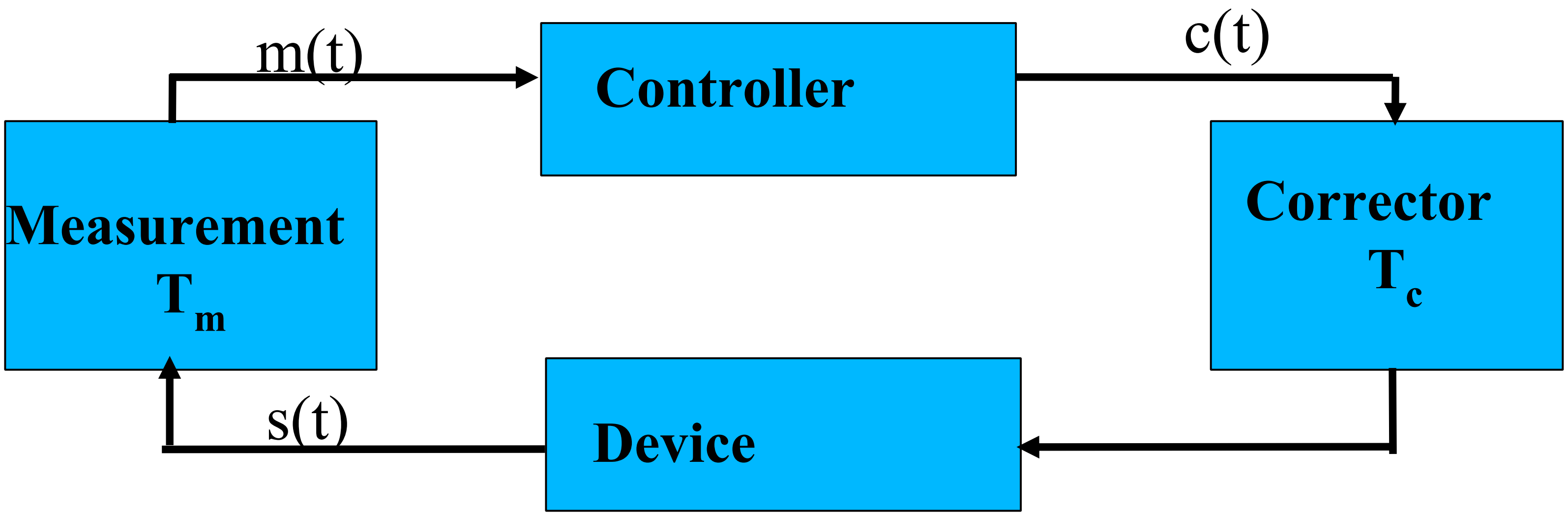
On Filter design...



Simplified view: Just regard delays

- At $t=t_0$, we see device state at $t=t_0-T_m$
- New corrector settings at $t=t_0$ will affect device only for $t> t_0+T_c$
- Now let's dream....

What if ....(the dream)



What if, using measurements of device state at $t = t_0$

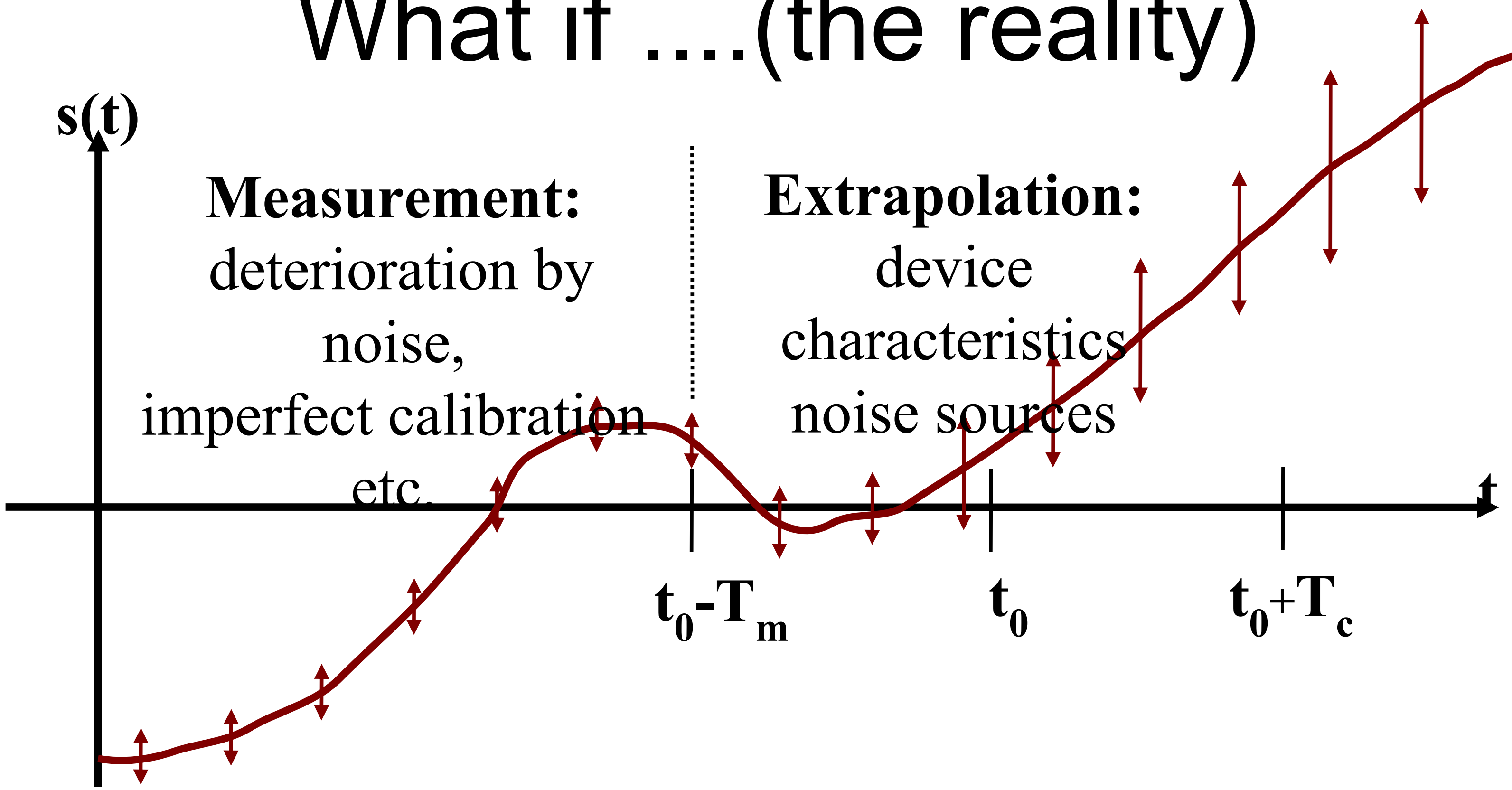
$$s(t),\ t < t_0 - T_m$$

we could perfectly predict device state $s(t = t_0+T_c)$?

- Setting correction at $t=t_0$ will affect device at $t = t_0+T_c$
- Trivial control problem: We can do perfect one step correction!

(but got noise, imperfect knowledge of elements etc....)

What if ....(the reality)



Goal:

Find process Ẽ, such that

$$\tilde{s}(t) = \tilde{E}\{s(t) \mid m(\tau), \tau < t - T_m - T_c\}$$

gives a best estimate of s(t)

Assuming linearity

Ansatz for the estimator

$$\tilde{s}(t) = \int m(\tau) h(t-\tau) d\tau$$

Minimizing the mean square error

$$P = E\left((s(t) - \tilde{s}(t))^2\right)$$

leads to:

$$R_{sm}(t,\zeta) = \int h(\tau) R_{mm}(\tau,\zeta) d\tau$$ (Wiener Filter)

with R_{sm} as the cross correlation function between s(t) and m(t)

(Integration limits decide, whether we are interpolating or extrapolating)

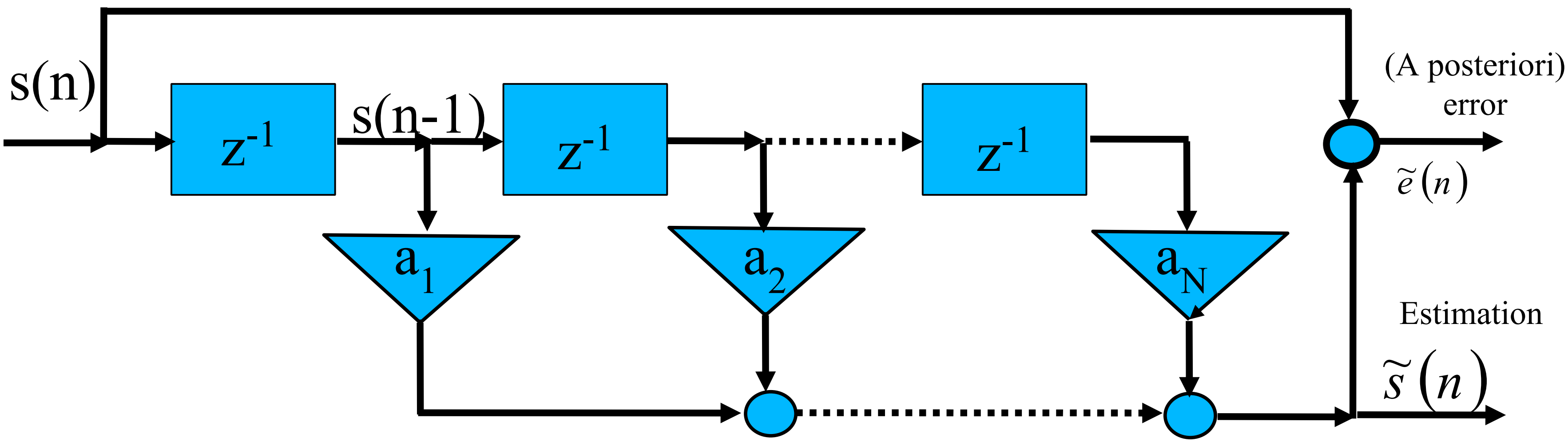
Ugly: Integral equation .....

A complication ...

- For closed feedback loop, filter may affect correlation matrices and complicate formulation and solution of problem. If yes:
  - Look for Wiener filter for open loop to give best estimates, add following standard controller to close loop (more comprehensible that way!)
  - (Youla Kucera parametrization for optimal controller design – if time permits, later)

Discrete FIR system

Measuring without noise: m(n)=s(m-1) , stationary noise $R_{xy}(t_1, t_2) = R_{xy}(t_1 - t_2)$



Now discrete problem of finite dimension, differentiation error function w.r.t. unknown filter coefficients gives N equation of the form

$$E\left\{\left(s(n) - \sum_{k=1}^{N} a_k s(n-k)\right) s(n-m)\right\} = 0, \quad 1 \le m \le N$$

or in terms of the autocorrelation: $R(m) - \sum_{k=1}^{N} a_k R(m-k) = 0, \quad 1 \le m \le N$

(Yule-Walker equations)

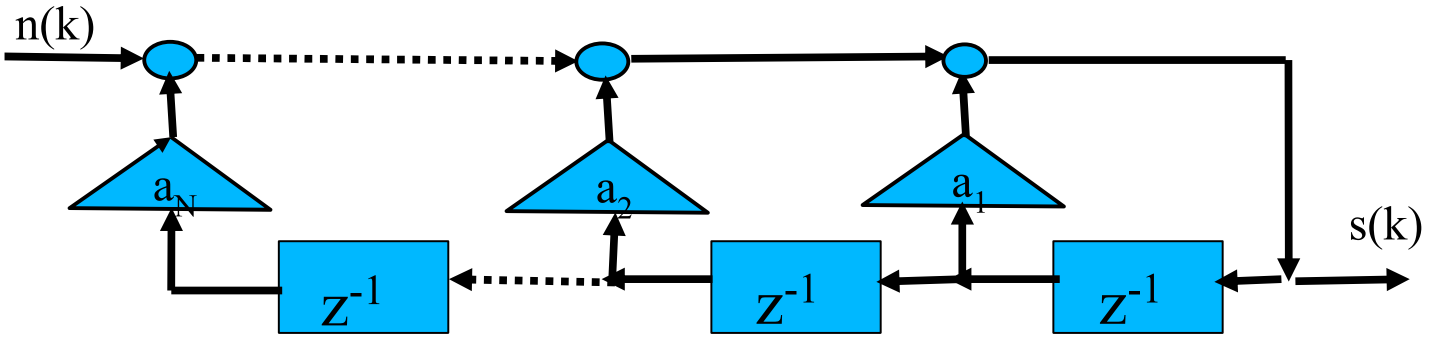
Use Levinson's recursion to find coefficients, specially for large N

(Matlab/Octave: durlev.m)

Features

•Filter looks for predictable part of the signal, so uncorrelated white noise as a signal results in a zero predictor!

•Assume, signal can be modeled as AR filter driven by white noise n(k)

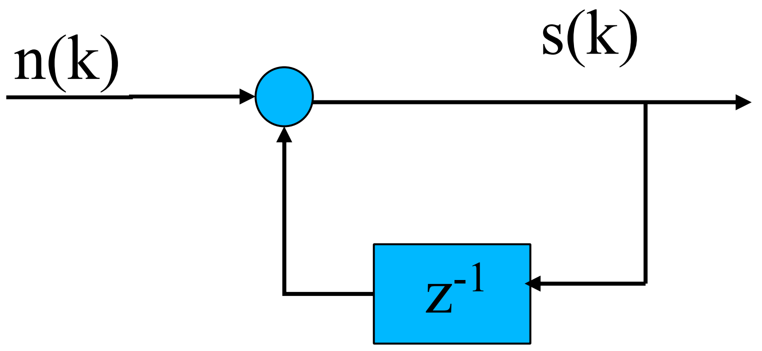


$$s(k)=\sum_{l=1}^{N}a_l s(k-l)+n(k)$$

Coefficients correspond directly to those of the (MA) Wiener filter:

$$\tilde{s}(k)=\sum_{l=1}^{N}a_l s(k-l)$$

Wiener filter for an ideal drift process



Integrator driven by white noise

$$s(k) = s(k-1) + n(k)$$

$$E(n(k)) = 0$$

$$R_{nn}(l) = E(n(k)n(k+l)) = \delta_l$$

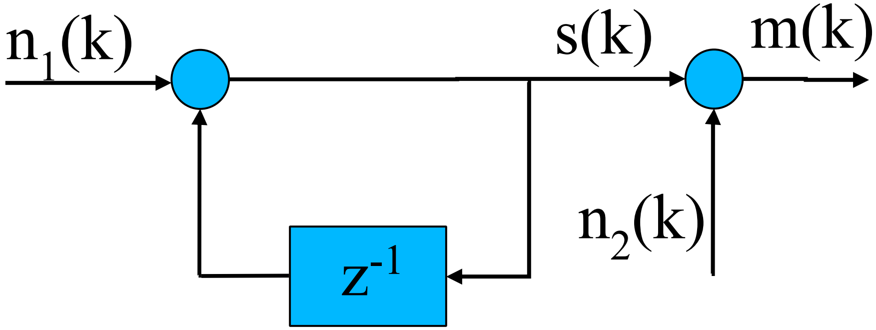
For a recursive AR process driven by white noise, the Wiener filter can be computed directly:

$$s(k)[\text{unknown}] = s(k-1)[\text{known}] + n(k)[\text{unknown}]$$

$$\rightarrow \tilde{s} = E(s(k)) = E(s(k-1) + n(k)) = s(k-1)$$

(Just the unity filter - boring, but true ...)

Ideal drift including measurement noise



Guessing state s(k) from measurements m(k):

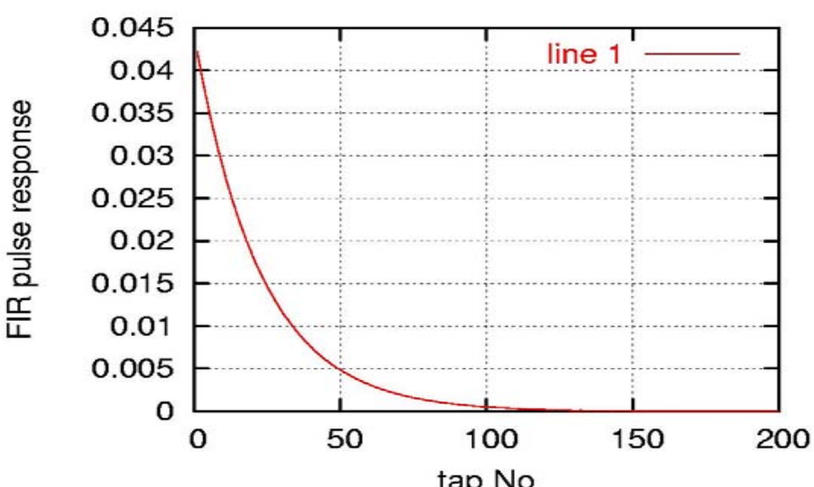
$$\tilde{s} = E(s(k)) = \sum_{l=1}^{N} a_l m(k-l)$$

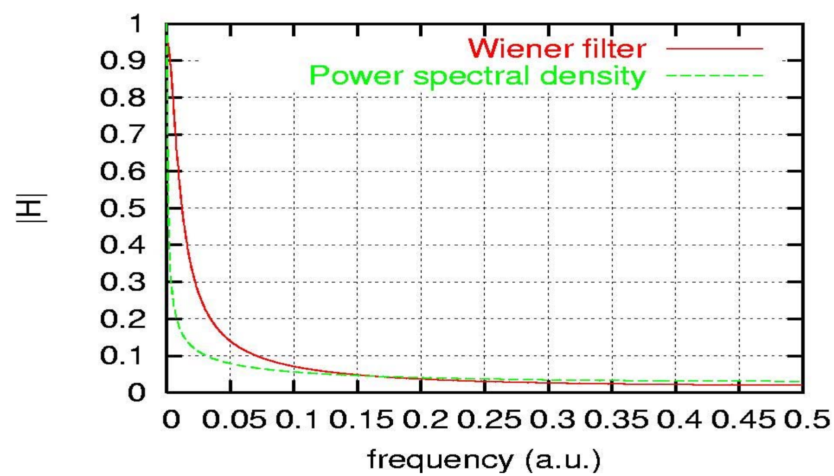
Least squares lead to modified Yule Walker equations:

$$\sum_{l=1}^{N} a_l R_{mm}(k-l) = R_{sm}(k); k = 1 \ldots N$$

Measurement noise n_2 white and uncorrelated to n_1

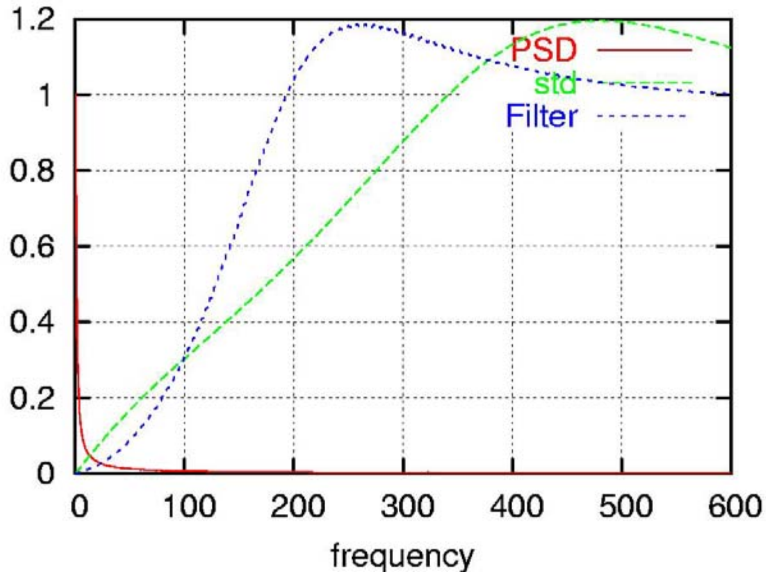
$$R_{mm}(k) = R_{ss}(k) + \delta_k ; R_{sm}(k) = R_{ss}(k)$$



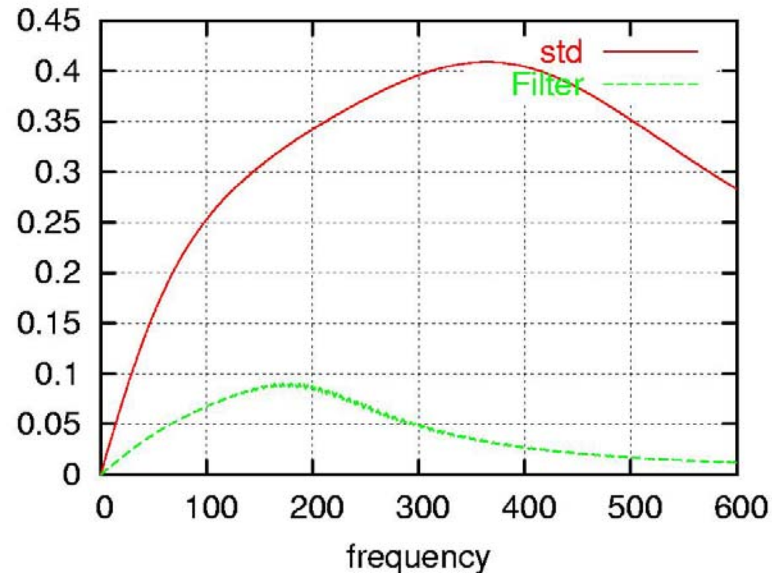


Typical closed loop behaviour (hand optimized)

Transferfct. Ground noise →orbit noise



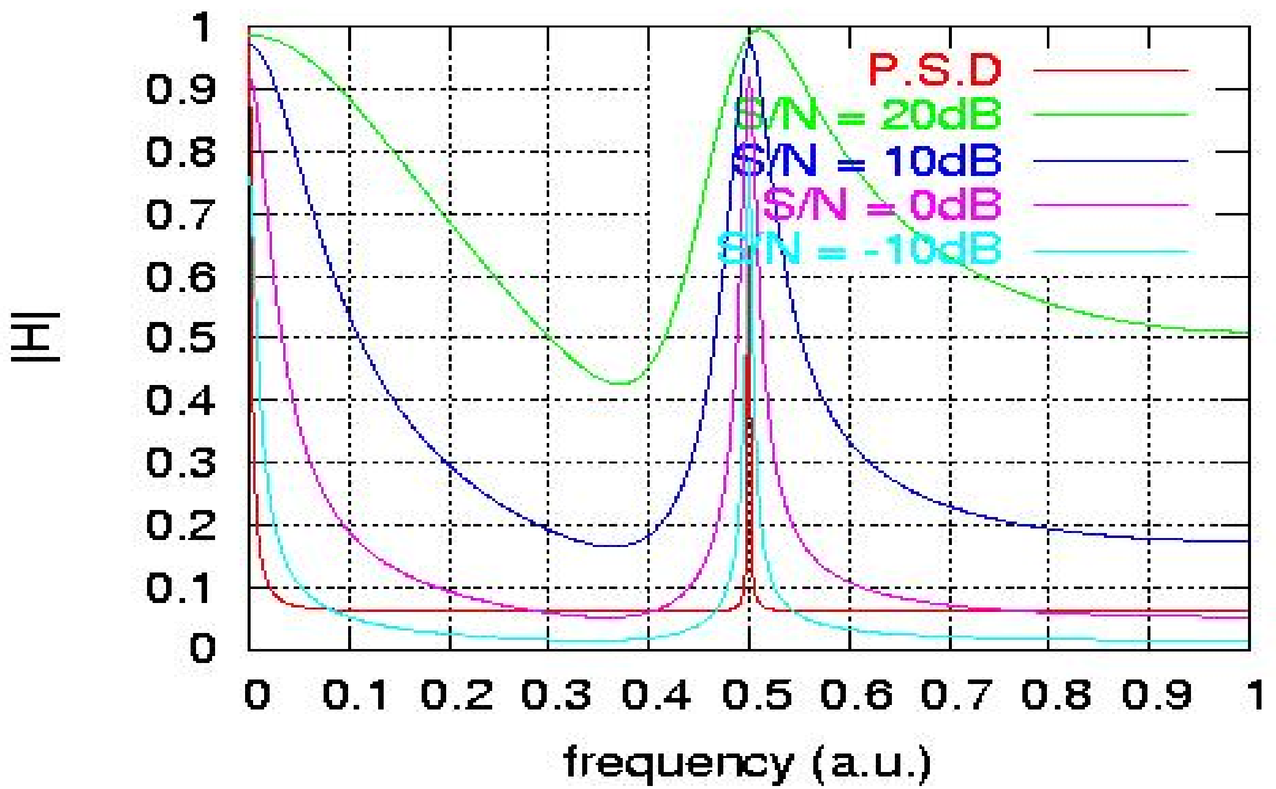
Transferfct. measurement noise →orbit noise



Balancing noise contributions!

Att: Minimizing fluctuations in estimation, but not measurement!

Drift + measurement noise + girder resonance

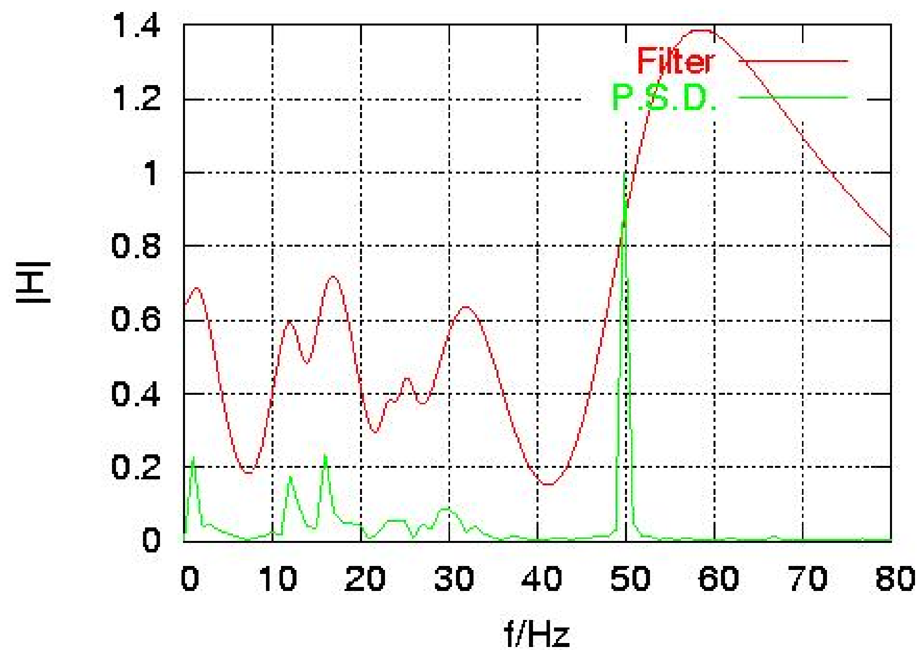


- Bad S/N ratios result in narrow band design
- Inherent compromise between latency (filter bandwidth) and low noise

A Wiener filter for a real spectrum (SLS orbit noise)

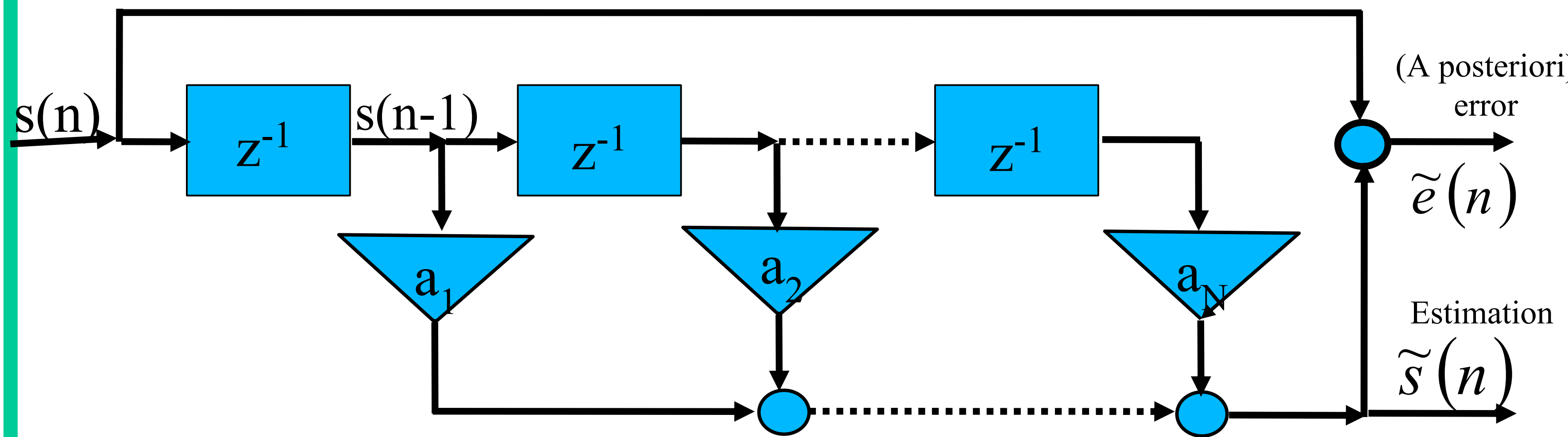
Direct approach only partially valid:

- Line at 50 Hz (power mains) not caused by system function, but systematic excitation!
  - Use Feed Forward or split frequencies via band pass to handle 50 Hz separately
- Line at 3 Hz is due to SLS booster cycle → same approach as for 50 Hz



Att: In practice, we see also spurious correlations and signals (e.g. from crosstalk) in the measurement – never confuse signal and measurement!

Quality criteria/adaptive filtering



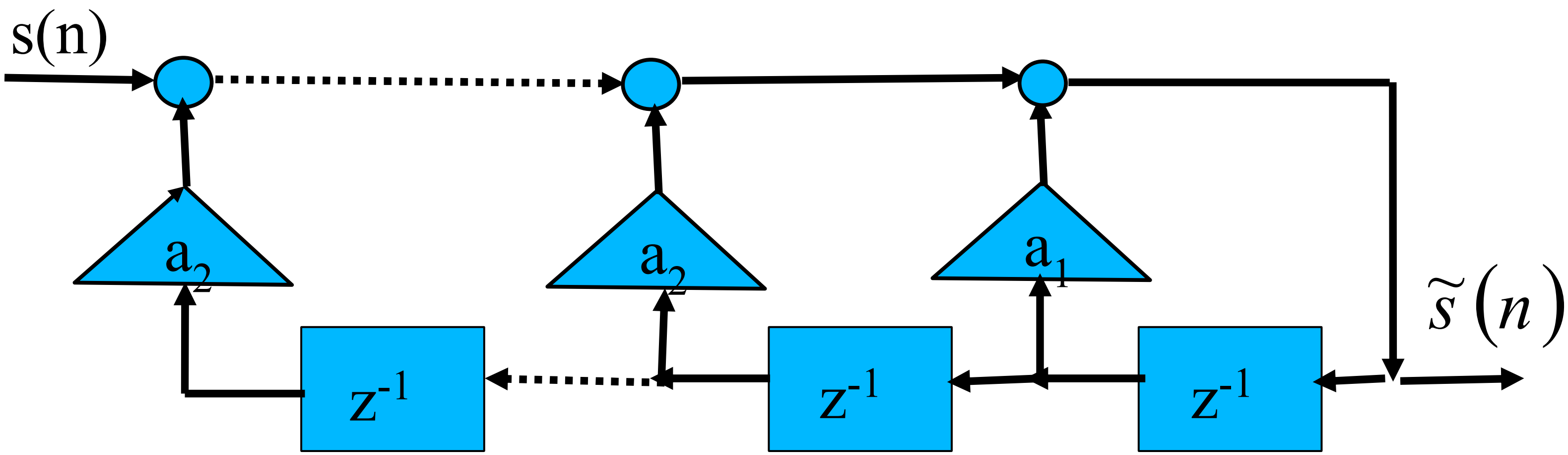
As the order of the filter tends to infinity, we get, the error does not go down to zero, but approaches the (unpredictable/uncorrelated) white noise: $R_{ee}(m) = K\,\delta_m$

Idea for adaptive filtering (non stationary noise spectrum):

Monitor autocorrelation of a posteriori error, if correlation gets too high:

- either recompute all signal correlations, recompute filter coefficients
- convert to Kalman filter:
  - Use coefficients $a_1 \ldots a_N$ and error correlation to compute new filter
  - Minimum interruption of filter operation during update

System description as IIR (AR) filters?



•Description of system eigen function as finite sum of complex exponentials and white noise

•Easy structure for narrow resonances in the spectrum

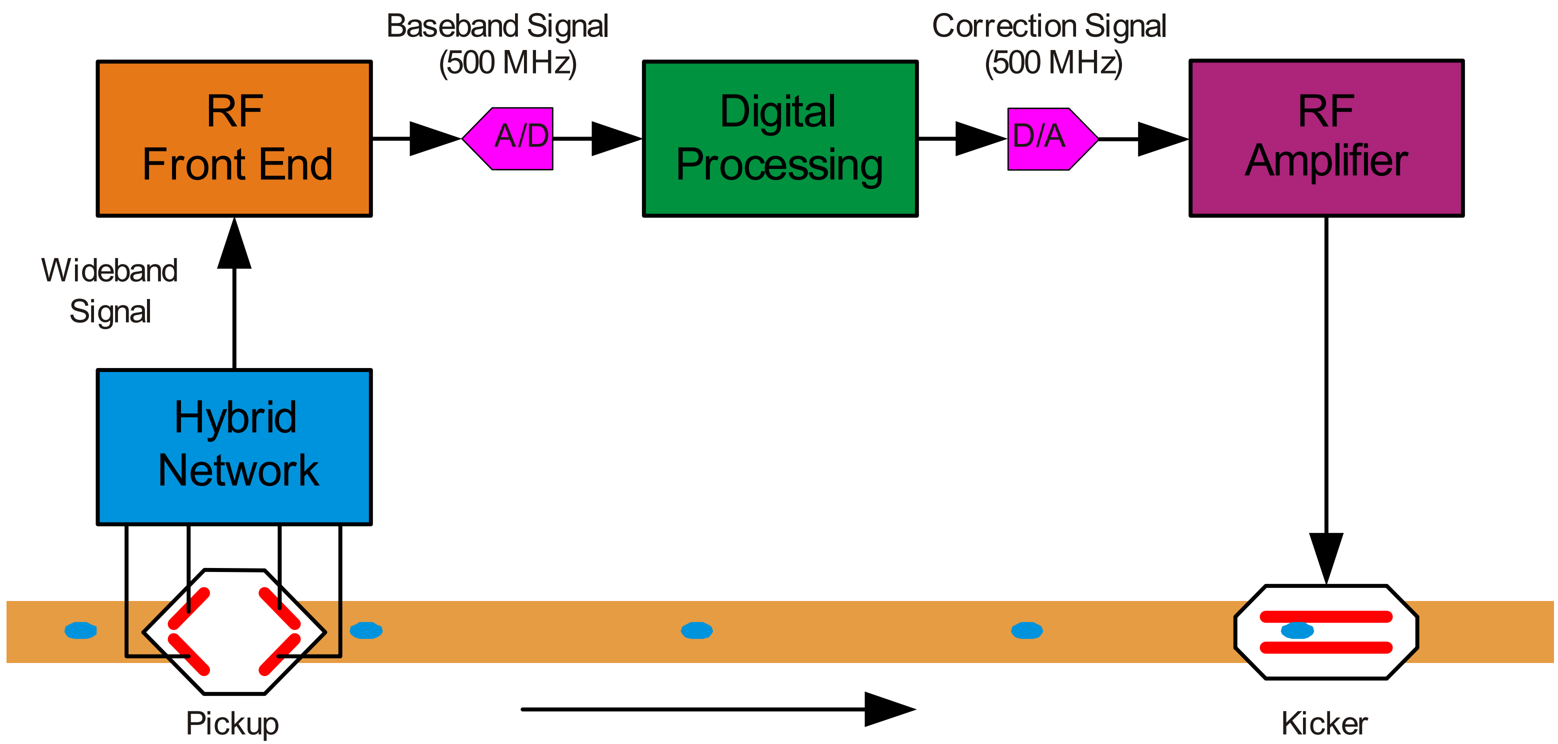
→ Pisarenko harmonic decomposition, resp.
Modified Prony Methods

•Need to know a priori number of resonances

A longitudinal
Bunch by Bunch feedback

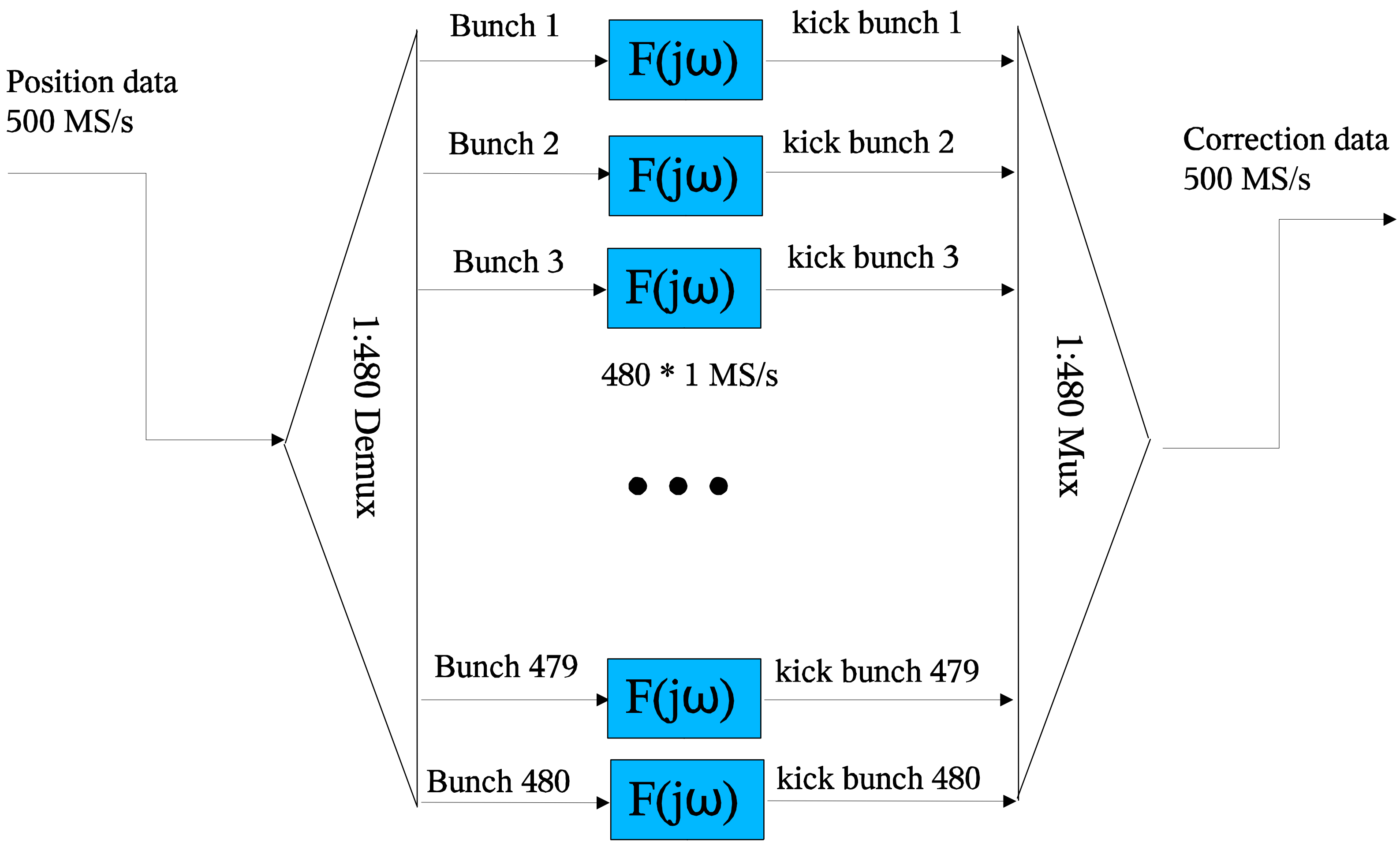
(seen from the signal processing view)

System layout

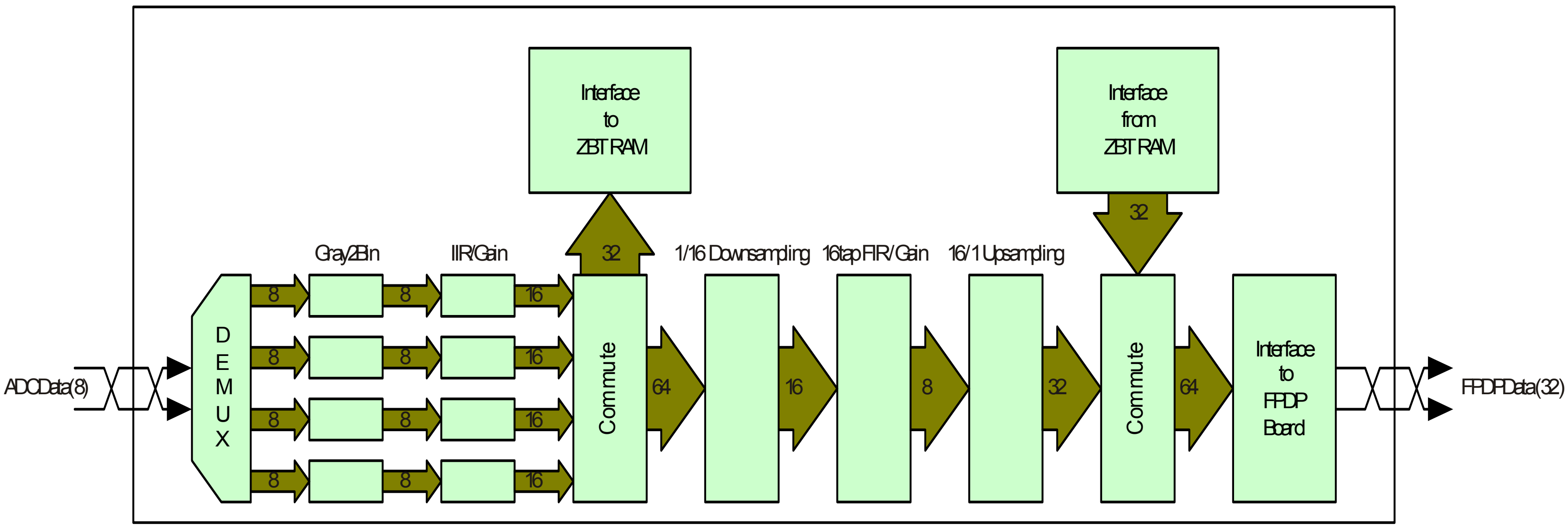


- Detailed description of this type of system see lecture by Marco Lonza
- Beam dynamics: Narrow band resonator at synchronous frequency/ies
- Analog components very wide band with negligible latency times
- How about signal and noise levels?

Bunch by Bunch filtering



Physical hardware layout of digital filter



Analog Noise

- Detector works at 1.5 GHz with a bandwidth of 500 MHz
- Thermal noise floor

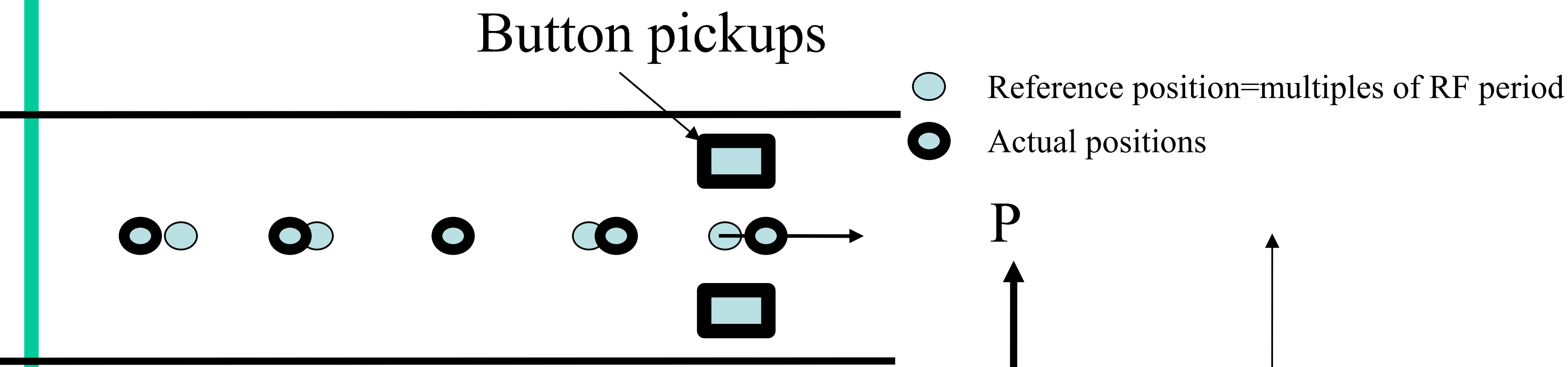
$$N = k_B TB = -77\,dBm$$

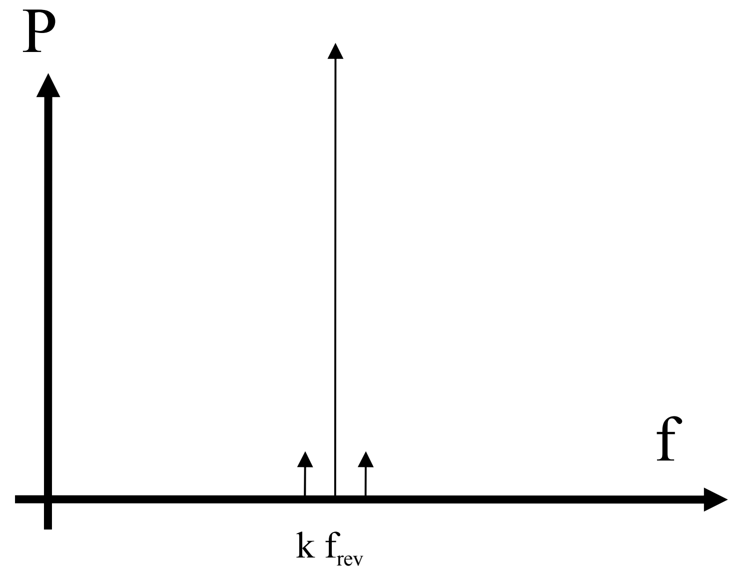
$$k_B = 1.4 * 10^{-23}\,Ws/K,\ \ T = 300\,K,\ \ B = 500\,MHz$$

Include estimated noise figure of RF front end of 7 dB gives

$$N = -70\,dBm$$

Signal levels





•Not interested in static offsets due to inhomogeneous fill patterns combined with transient beam loading effects

•Longitudinal oscillations create side bands of revolution harmonics (phase modulation of beam signal)

•beam signal: $$V(t) = Ae^{j(\omega t+\phi \cos \omega_s t)}$$

•For small modulation amplitudes φ $$Ae^{j(\omega t+\phi \cos \omega_s t)} = Ae^{j\phi \cos \omega_s t} e^{j\omega t} \approx A(1 + j\phi \cos \omega_s t)\, e^{j\omega t}$$

Side bands at $\pm \omega_s$ with relative amplitude φ/2

Main peak: -1 dBm at 400 mA and noise floor of -71 dBm result in 0.3 µrad phase noise

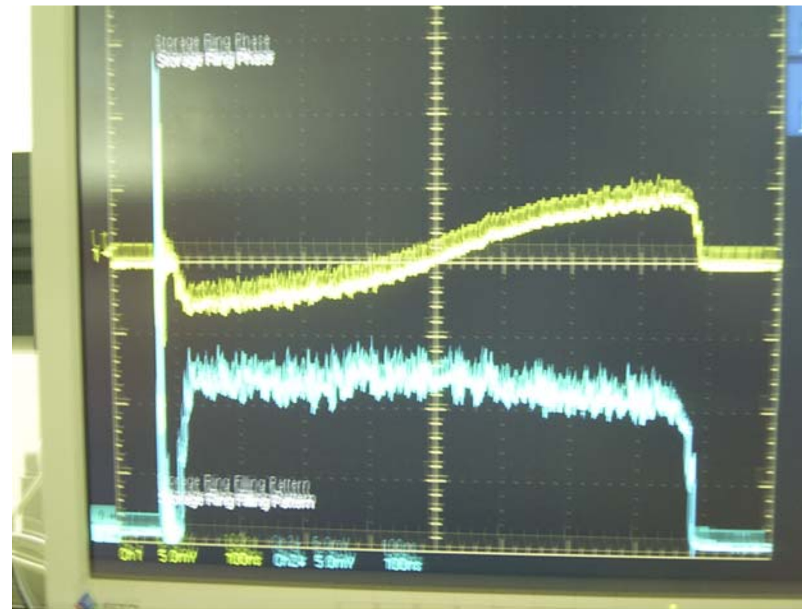
Measuring around 1.5 GHz gives resolution of 0.3 µrad*660ps = 2 ps (~ 1.2 keV)!

Analog/Digital Conversion

- Adjusting signal level to ADC range gives optimum S/N ratio to be expected
  - 8 bit ADC with nominally 6.5 bit resolution
  - Best S/N = 6.5 bit * 6 dB/bit

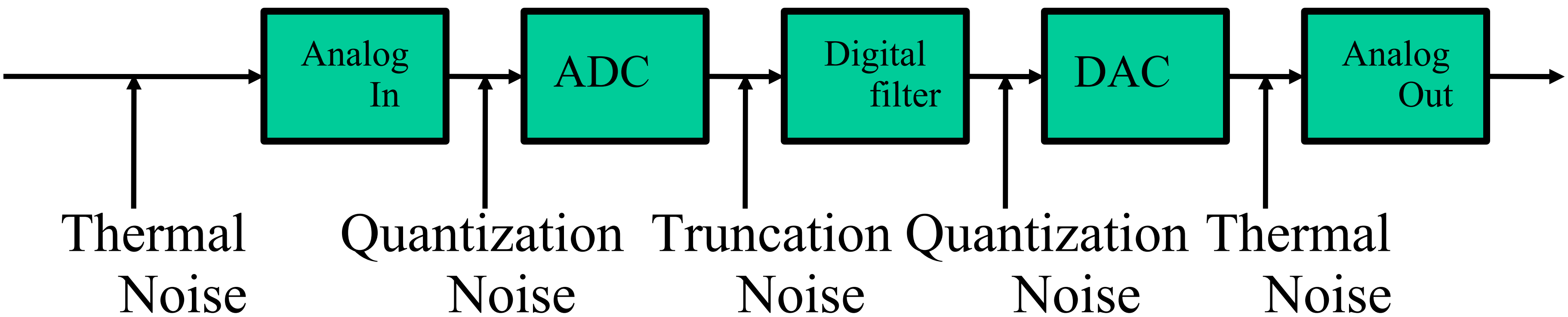
    S/N = 39 dB

  BUT
- Typically transient beam loading give shift in synchronous phase
- Need to waste lots of ADC resolution (8 bit) to capture transient beam loading effects.



- Blue trace: Fill pattern
- Yellow trace: Phase lag due to transient beamloading

Overview of inherent noise sources



- Analog input noise small compared to ADC quantization effects.
- Truncation noise → see later
- White noise from DAC and Analog output circuitry no problem, since the beam constitutes band pass at synchronous frequency, sees only small fraction.
- Big problem: ADC noise may saturate Digital filter/DAC/Power amplifiers etc.
- How to set up the filter (Algorithms, data flux and format)?

How to recover resolution

- Only way of cutting noise power after ADC is reducing bandwidth at frequencies off synchronous frequency
- In principle need only band of few hundred Hz around synchronous frequency (2-5 kHz)
- Noise power is spread out within 0-500 kHz (for a single bunch signal) including also lines at multiples of 50 Hz, crosstalk from RF FB loops, betatron motion etc.
- Use bandpass around synchronous frequency to cut the noise without affecting the signal (also avoid aliasing)!
- Divided functions: Introductory general purpose bandpass as an IIR filter combined with configurable FIR controller.
- Try to do computation, while generating minimum noise

Side remark on data types

It may be tempting trying to design filters in floating point format. With this, one incurs two problems.

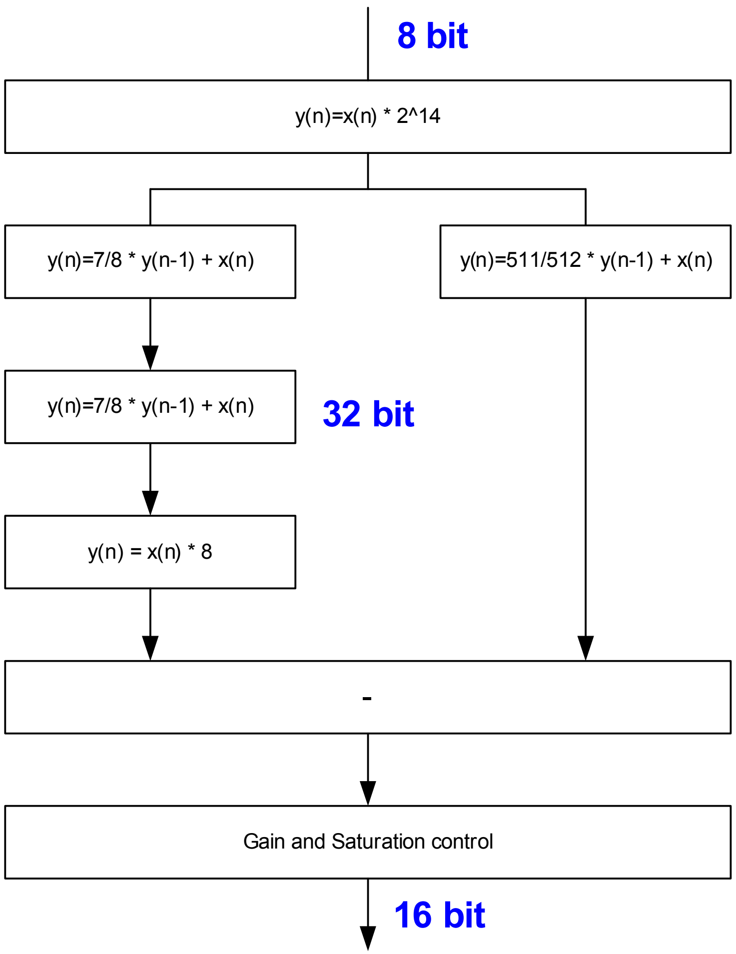
Implementation may be inefficient in terms of computing time and resources and on e.g. FPGA even impossible.

A 32 bit float has a mantissa of only 23 bit, which gives the relative resolution of the format, and so inherently is more ‚noisy' than a fixed point value of the same size.

Use fixed point/integer and do gain adjustments to make best use of available resources.

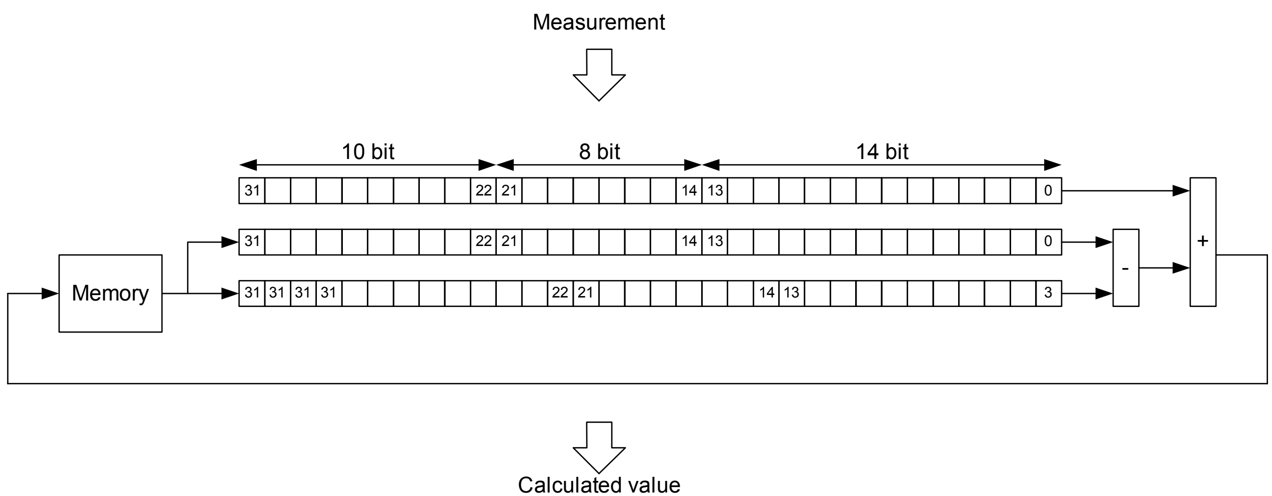
- A data format/type in itself does not have a S/N ratio/noise figure
- Mathematical operation and their structure (**truncation**) creates noise and decreases S/N
- Examples:
  - a [8 bit] * b [8 bit] → c [16 bit] (No noise)
  - (a [8 bit] + b [8 bit]) /2 → c [8 bit] (Truncation noise 2^{-9})
  - a [FP] + b [FP] → c [FP] (Noise: 2^{-24} * max(a,b))
  - a [FP] * b [FP] →c [FP] (Noise: 2^{-24} * c)

Introductory IIR filter



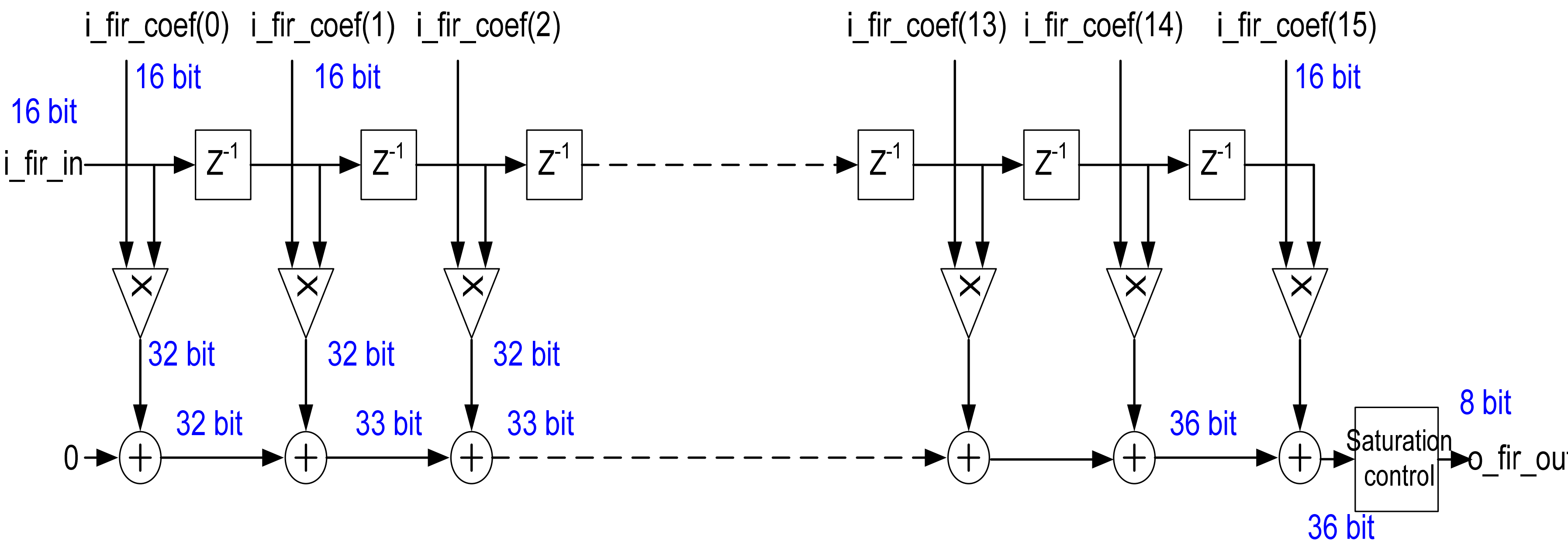
- Expand input data to 32 bit
- Band pass (0.3-20 kHz) as difference of two low pass
- reduction to 16 bit at end

Doing first stage of IIR with maximum precision (minimum noise)



- IIR: $y_n = 7/8\, y_{n-1} + x_n$
- Realized via shifts and adds
- Output has significant bits in fields 0 to 24
- Second IIR significant bits in fields 0 to 27
- Multiplier stage (factor 8 fills to fields 0 to 30)

FIR filter stage



- User configurable coefficients (16 bit width)
- Variable data formats to avoid saturation and loss of precision
- Saturation and control stage at end reduces data to 8 bit width suitable for digital to analog conversion

Choosing FIR filter coefficients

(the simple way)

- Need to choose 16 coefficients (A System from Tokyo Electron Devices has even 64 coefficients..)
- Nonstandard system: We want to correct for non zero synchrotron frequency
- What does a PID controller design for non DC application look like at all?

PID at DC | centered at ω_0

Proportional

$$F(j\omega) = a$$

$$F(j\omega) = a$$

Integrator

$$F(j\omega) = \frac{b}{j\omega}$$

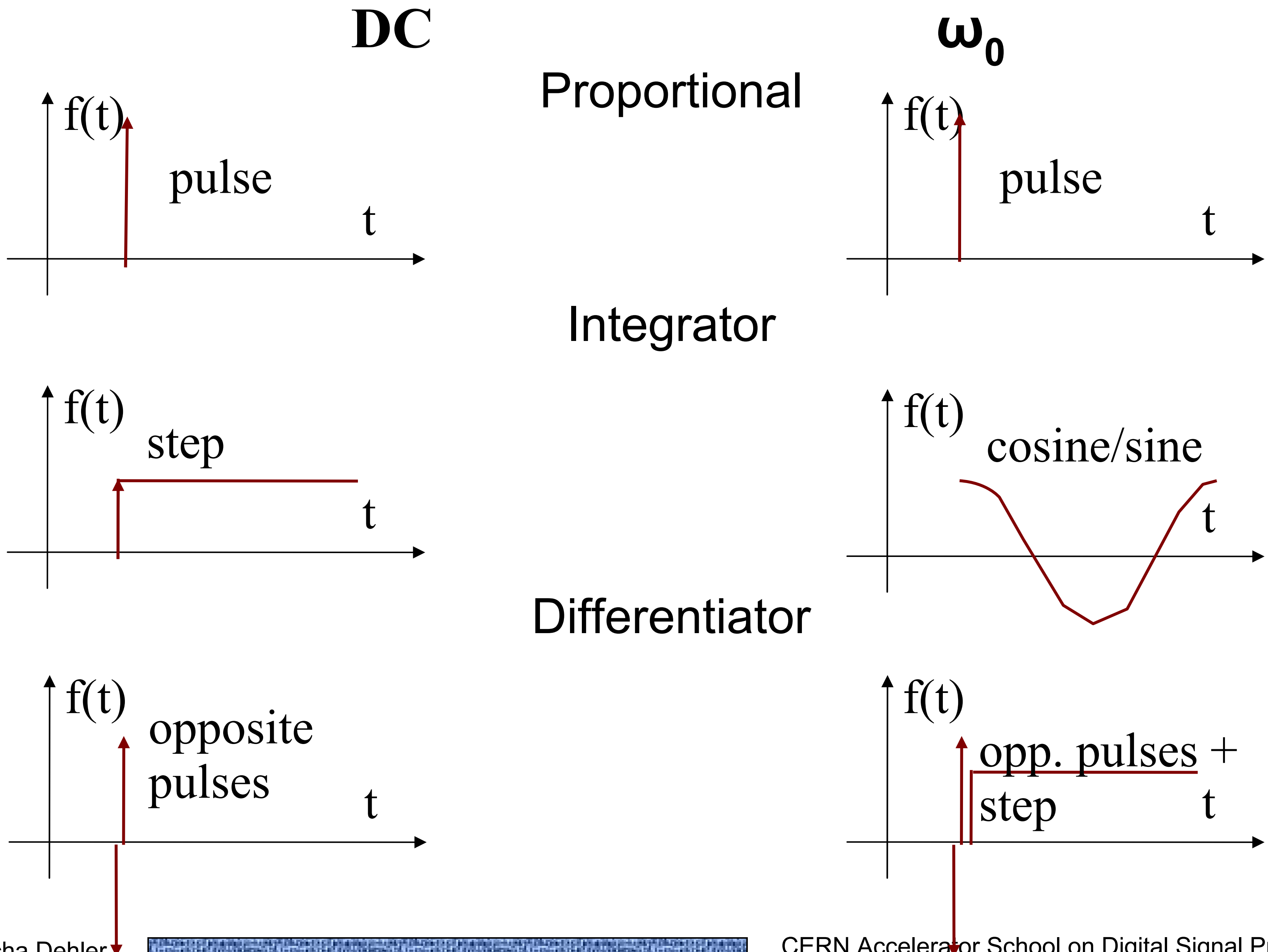
$$F(j\omega) = \left(b_r + jb_i\frac{\omega_0}{\omega}\right)\frac{2j\omega}{\omega^2 - \omega_0^2}$$

Differentiator

$$F(j\omega) = j\omega$$

$$F(j\omega) = \left(c_r + jc_i\frac{\omega}{\omega_0}\right)\frac{\omega^2 - \omega_0^2}{2j\omega}$$

Comparing the pulse responses



The recipe

- Shift classical PID controller to design frequency ω_0
- Compute pulse response of controller
- FIR coefficients describe function at discrete times

$$a_i = h(t=i\ T)$$

(or do FIR/IIR design)

- Got to choose and adjust now 5 numbers (constant for P, real & imaginay part for I and D) instead of 3 for the DC PID ...

Low level architecture on a FPGA

- Basic Adder and Multiplier block
- (not used exactly like this in the system)

- Variable FIR/IIR through configurable data switches, delays

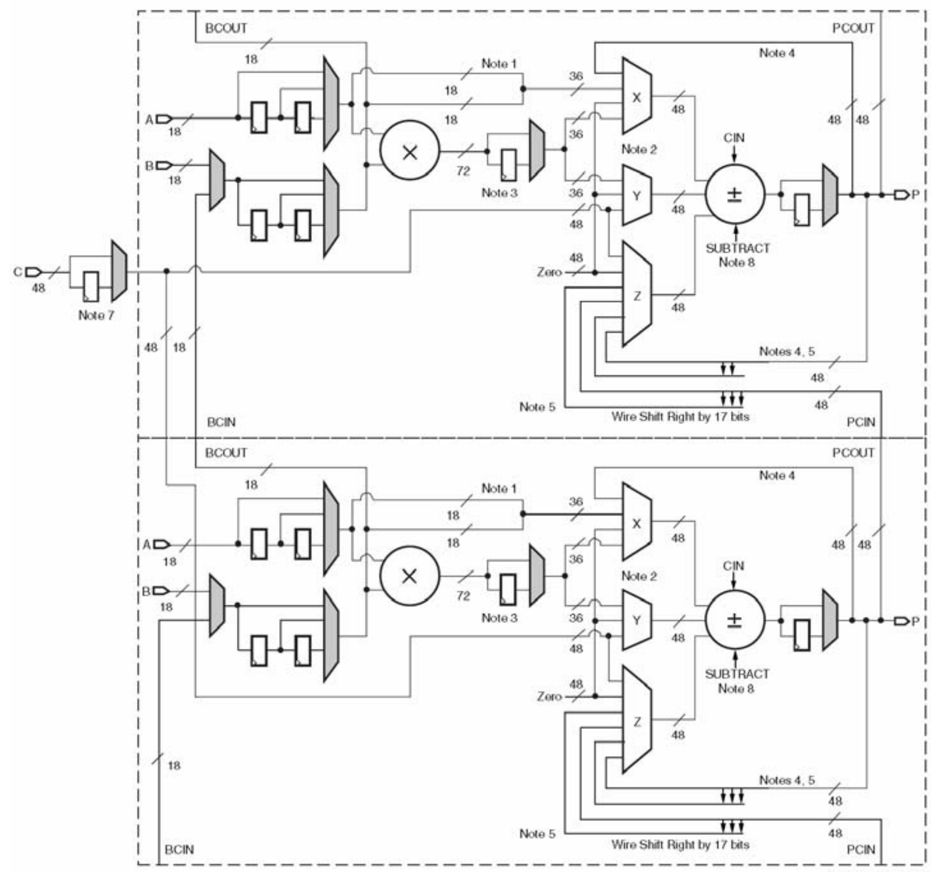
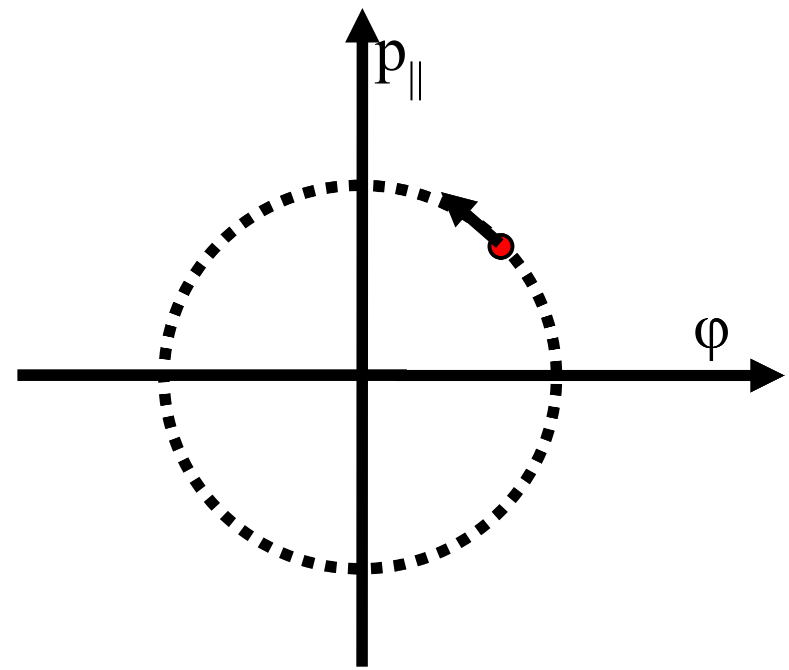


Figure 1-3: **A DSP48 Tile Consisting of Two DSP48 Slices**

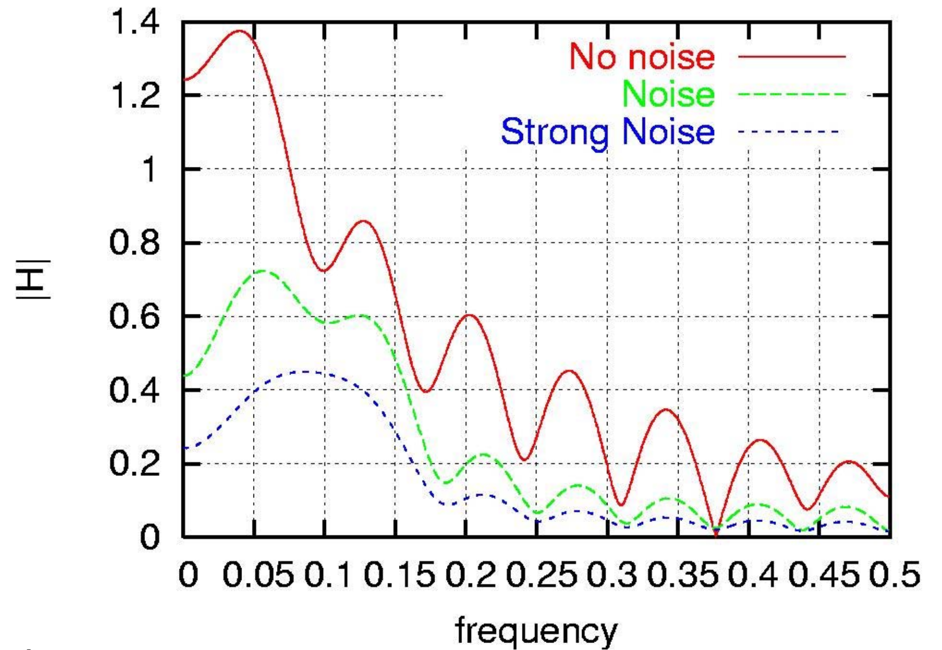
Wiener filter for a bunch by bunch feedback



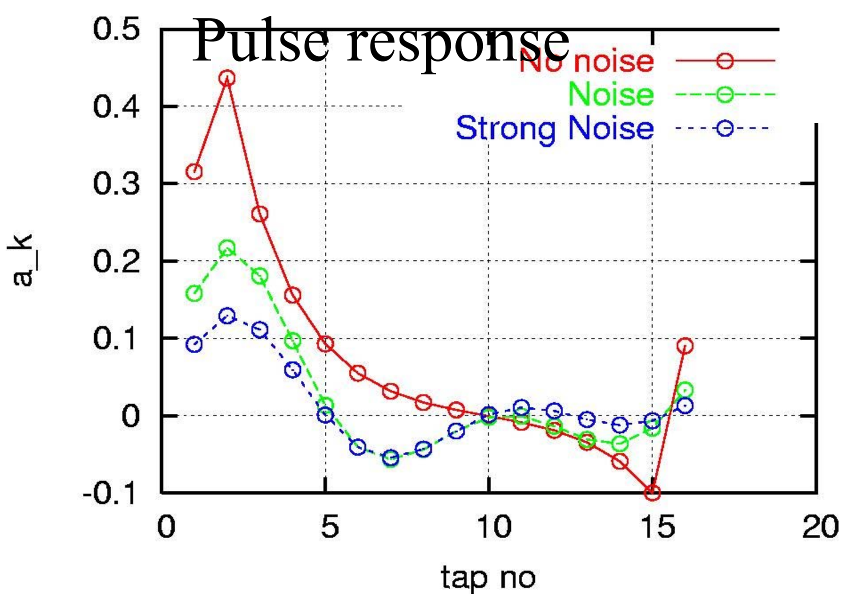
- Bunch perform oscillatory motion in longitudinal phase space
- Spectral density/autocorrelation described by resonance with positive/negative damping coefficients
- **Goal not to suppress fluctuations, but to shift instable poles!**
- Can use filter directly as controller
- **We measure** the longitudinal offset of the bunches w.r.t. the reference phase [= m(k)]!
- **We need to know** and correct the longitudinal momentum [= s(k)]
- Complex phase shift 90 degree between m(k) and s(k)
- Synthesizing filter from theoretical model ....

16 tap filters w/wo noise

- Equations for N=16 coefficients →other as ideal Wiener N→∞
- Design frequency fs=0.1
- In the absence of noise, filter is just guessing for right phase
- With increasing noise converging to band pass



Pulse response



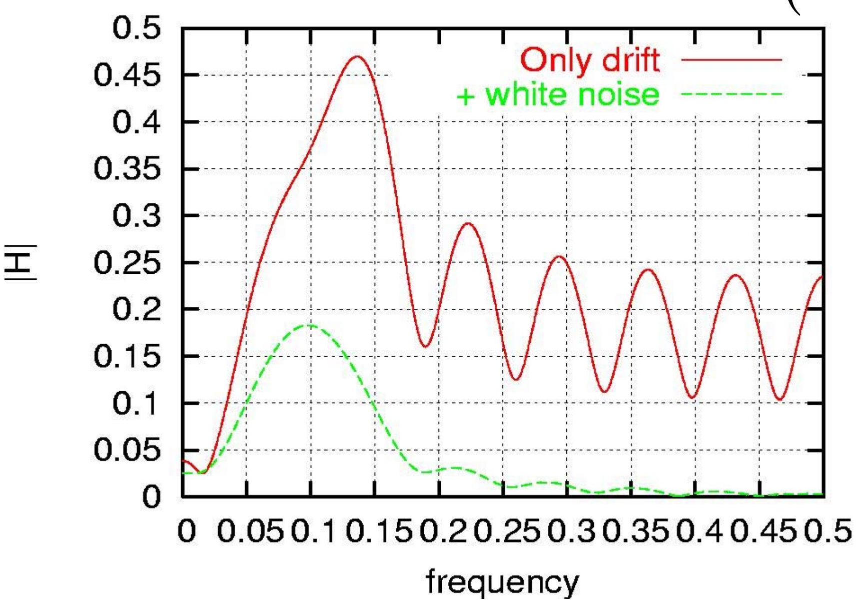
Static offsets/spurious drifts

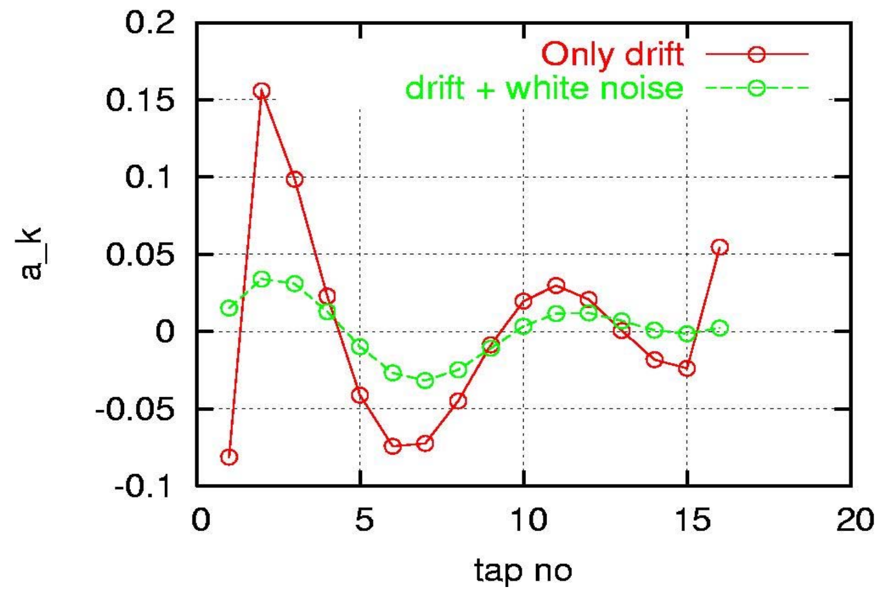
- In principle measured signal also contains a drifts/offsets

$$S_m(j\omega) = \underbrace{\frac{A\omega^2}{\left(\omega^2 - \omega_s^2\right)^2}}_{\text{synchrotron oscillation}} + \underbrace{\frac{B}{\omega^2}}_{\text{drift}} + \underbrace{N}_{\text{white noise}}$$

- To isolate oscillatory signal, exclude drift (analytically?)from signal spectrum/correlation function

$$S_s(j\omega) = \frac{A}{\left(\omega^2 - \omega_s^2\right)^2}$$





Tolerances and imprecisions

- Have been talking about incorporating missing knowledge about signal into filter design.
- How about other questions:
  - Nonstationary noise
  - How good is the physical model?
  - Nonstationary devices
- Best answer to that typically Kalman filter, but
  - Need to know true signal (as opposed to measurement)
  - Need to know true cross correlations
  - Uneasy feelings on effect on system stability..
- Other ways to approach problem?

the practical problem ..

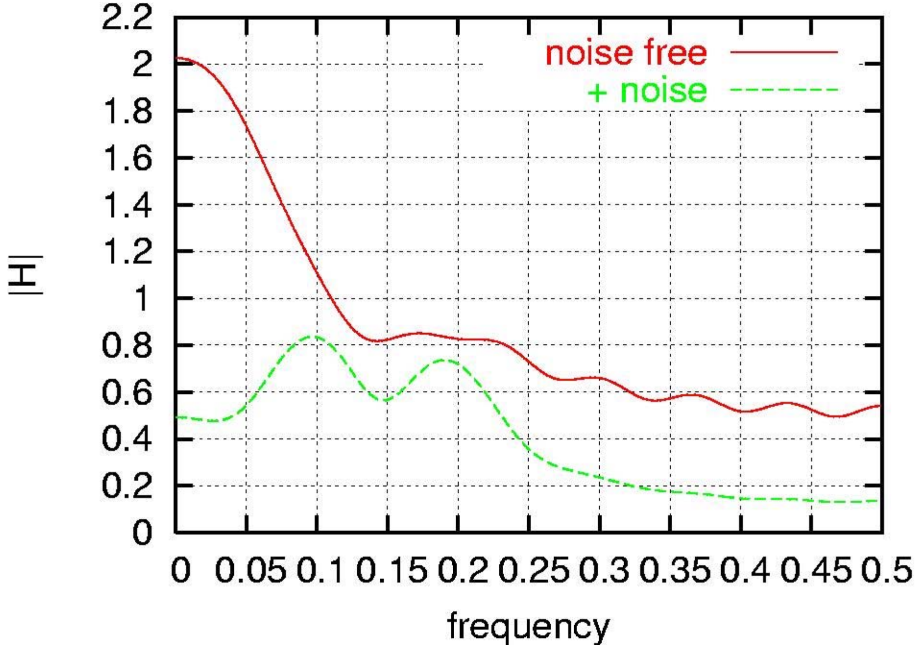
- SLS has a superconducting third harmonic cavity. It is a passive device driven by beam helping to produce long bunches and increasing stability thresholds by changing the slope of RF voltage versus phase.
- As side effect, get change in synchronous frequency vs. voltage (that is, also vs. beam current) in the range 2-5 kHz
- As side effect, synchronous oscillations/instabilities affect also cavity voltage
- The vicious circle (with a standard controller filter):
    1. Oscillation starts
    2. Voltage in cavity drops
    3. Change in synchronous frequency
    4. Feedback becomes more inefficient, because filter no more optimum
    5. Oscillation amplitude increases
    6. Goto 2

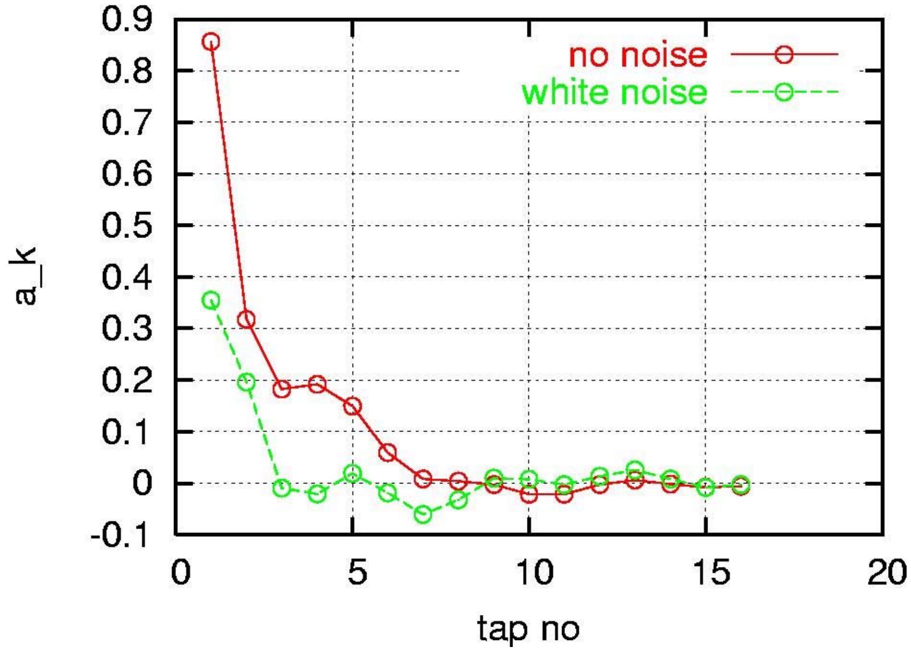
Solution: Include the expected range of the synchronous frequency (as well as other variable) as a stochastic density into the design process

... a double stochastic filter

- Optimize for frequencies between 0.1 and 0.2
  - Density of f_s: $$D(f_s) = \begin{cases} C; 0.1 < f_s < 0.2 \\ 0; else \end{cases}$$
  - Resultant PSD:

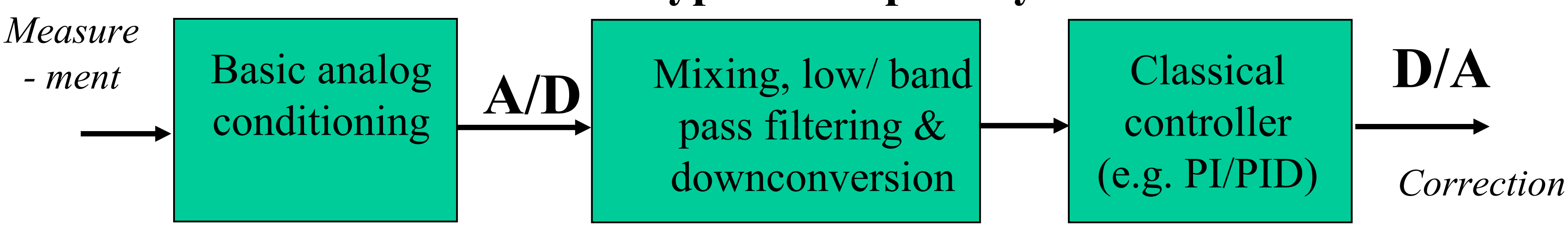
$$S_s(j\omega) = \int S(\omega, f_s) D(f_s) df_s = \int \frac{A\omega^2}{\left(\omega^2 - (2\pi f_s)^2\right)^2} df_s$$





Where to go from here

Typical setup today



- We will be able to process faster...
- We will be able to process more...
- For current typical setup:
  - Shorter latency
  - Less internally generated noise due to extended data formats
- Use of optimal controllers: Wiener/Kalman/LQR etc. (Understand stochastic properties of your processes!)
- Replace even digital receiver by predictor?

Recapitulation

- “Real time systems”
- Orbit feedback
    - Static physical model
    - Dynamics
    - Fundamental properties
    - Wiener filters I
        - Design for generic spectra
        - Real spectra: limits to design process
        - Other options

Recapitulation II

- Youla Kucera parametrization of stable controllers
- Bunch/Bunch Feedback
  - Physical model
  - Noise
  - Global filter layout
  - High speed filter computations with minimum noise
  - Adapted PID controllers
  - Wienerfilter II
    - Adapted design approach
    - Designs for generic spectra
    - Incorporating model tolerances into filter design