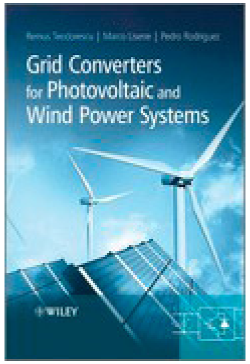


Grid Converters for Photovoltaic and Wind Power Systems

by R. Teodorescu, M. Liserre and P. Rodriguez

ISBN: 978-0-470-05751-3

Chapter 4

Grid Synchronization in Single-Phase Power Converters

Outline

- Introduction
- Grid synchronization by using the Fourier analysis
- Grid synchronization by using a Phase-Locked Loop
- Phase detection based on in-quadrature signals
  - PLL based on a T/4 transport delay
  - PLL based on the Hilbert transform
  - PLL based on the inverse Park transform
- Adaptive filtering
  - The Enhanced PLL
  - Second order adaptive filter
- Second order generalized integrator
- The SOGI-PLL
- The SOGI Frequency-Locked Loop
- Conclusions

Introduction

- The power converter of renewable energy systems should accurately screen the grid variables at the point of common coupling in order to trip the disconnection procedure when they go beyond of the limits set by the grid codes
- Grid synchronization is an adaptive process by means of which an internal reference signal generated by the control algorithm of a grid-connected power converter is brought into line with a particular grid variable, usually the fundamental component of the grid voltage
- Knowing the magnitude and phase-angle of the grid voltage allows regulating the active and reactive power delivered to the grid by a grid-connected power converter
- The frequency-domain synchronization methods are usually based on any discrete implementation of the Fourier analysis.
- The time-domain detection methods are based on some kind of adaptive loop that enables an internal oscillator to track the component of interest of the input signal

Grid synchronization by using Fourier analysis

- Fourier Series:

  Fourier stated that a generic periodic signal v(t) can be expressed by a sum of the following terms:

$$v(t) = a_0 + \sum_{n=1}^{\infty}\left(a_n \cos(n\omega t) + b_n \sin(n\omega t)\right)$$

where:

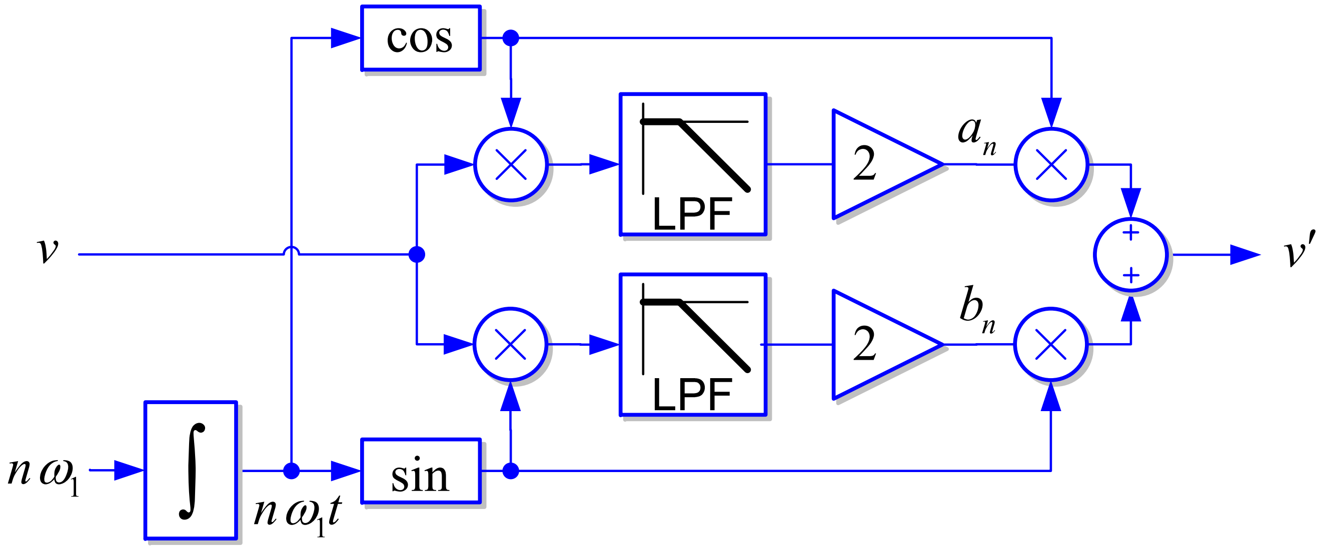
$$a_0 = \frac{1}{T}\int_0^T v(t)\, dt,$$

$$a_n = 2\frac{1}{T}\int_0^T v(t)\cos(n\omega t)\, dt = \frac{1}{\pi}\int_{-\pi}^{\pi} v\left(\theta\right)\cos(n\theta)\, d\theta,$$

$$b_n = 2\frac{1}{T}\int_0^T v(t)\sin(n\omega t)\, dt = \frac{1}{\pi}\int_{-\pi}^{\pi} v\left(\theta\right)\sin(n\theta)\, d\theta.$$

Grid synchronization by using Fourier analysis

- Adaptive filter based on Fourier series decomposition



$$\vec{V}_n' = V_n \angle \theta_n \begin{cases} V_n = \sqrt{a_n^2 + b_n^2} \\ \theta_n = \arctan \dfrac{b_n}{a_n} \end{cases}$$

Grid synchronization by using Fourier analysis

- Complex form of the Fourier Series

  The Fourier series' coefficients can be also calculated by

$$a_n = \frac{1}{T}\int_0^T v(t)\left(e^{jn\omega t} + e^{-jn\omega t}\right)dt,$$

$$b_n = \frac{-j}{T}\int_0^T v(t)\left(e^{jn\omega t} - e^{-jn\omega t}\right)dt$$

  Therefore, defining the complex coefficient c_n as:

$$c_n = \frac{1}{2}\left(a_n - jb_n\right) = \frac{1}{T}\int_0^T v(t)e^{-jn\omega t}dt$$

  and the Fourier series of $v(t)$ can be rewritten as:

$$v(t) = a_0 + \sum_{n=1}^{\infty} c_n e^{jn\omega t} + \sum_{n=1}^{\infty} c_n e^{-jn\omega t} = \sum_{n=0}^{\infty} c_n e^{jn\omega t} + \sum_{n=-1}^{-\infty} c_n e^{jn\omega t} = \sum_{n=-\infty}^{\infty} c_n e^{jn\omega t}$$

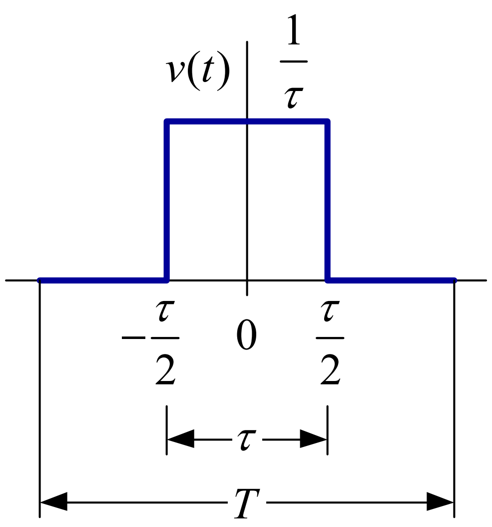
Grid synchronization by using Fourier analysis

- Fourier Transform (FT)

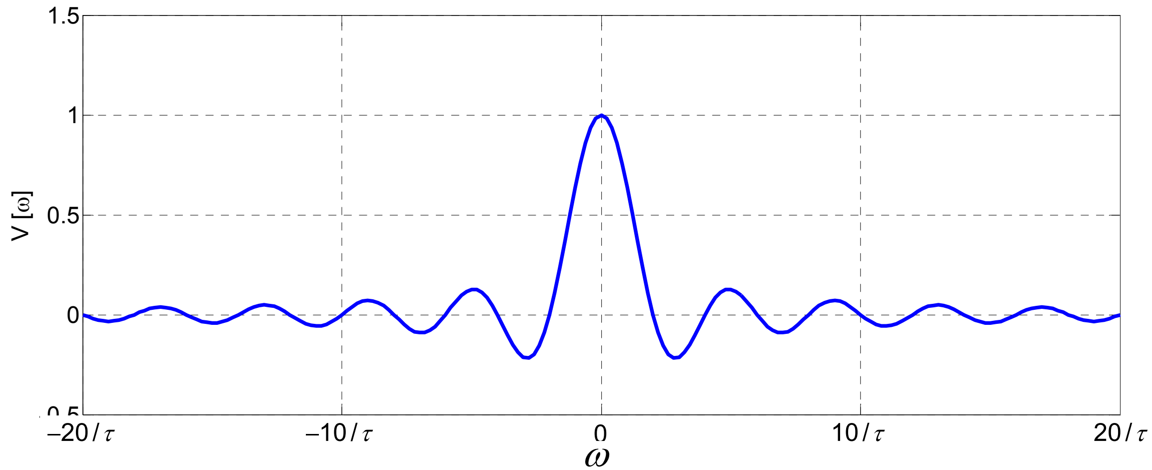
  The Fourier transform allows applying the time/frequency duality in the analysis of aperiodic signals.

$$V(\omega) = \mathcal{F}\left[v(t)\right] = \int_{-\infty}^{\infty} v(t) e^{-j\omega t} dt$$

Time domain



Frequency many



Grid synchronization by using Fourier analysis

- Discrete Fourier Transform (DFT)

  Defining the discrete input signal is defined as:

$$v[k] = v(t)\cdot\delta\left(t - kT_S\right)\ ;\ k = 0,1,\ldots.,N-1$$

where $d(x)$ is the Dirac's delta function used for sampling, T_S is the sampling period, and N is the number of samples to be processed.

and replacing integrals by summation of a finite number of samples equally spaced in time, the Discrete Fourier Transform (DFT) is defined by:

$$V[n] = \sum_{k=0}^{N-1} v[k]\cdot e^{-j2\pi\frac{k}{N}n}$$

and the inverse discrete Fourier transformation (IDFT) is defined as:

$$v[k] = \frac{1}{N}\sum_{n=0}^{N-1} V[n]\cdot e^{j2\pi\frac{n}{N}k}$$

Grid synchronization by using Fourier analysis

- Recursive Discrete Fourier Transform (RDFT)

By computing the DFT algorithm at the $[k_S\text{-}1]$ and $[k_S]$ samples to extract the n^{th} harmonic of the input signal we have:

$$V[n]\Big|_{k_S-1} = \sum_{k=k_S-N}^{k_S-1} v[k]\cdot e^{-j2\pi\frac{k}{N}n} \qquad V[n]\Big|_{k_S} = \sum_{k=k_S-N+1}^{k_S} v[k]\cdot e^{-j2\pi\frac{k}{N}n}$$

Subtracting both equation and simplifying, the Recursive Discrete Fourier Transform (RDFT) algorithm can be formulated as:

$$V[n]\Big|_{k_S} = V[n]\Big|_{k_S-1} + v[k_S]\cdot e^{-j2\pi\frac{k_S}{N}n} - v[k_S-N]\cdot e^{-j2\pi\frac{k_S-N}{N}n}.$$

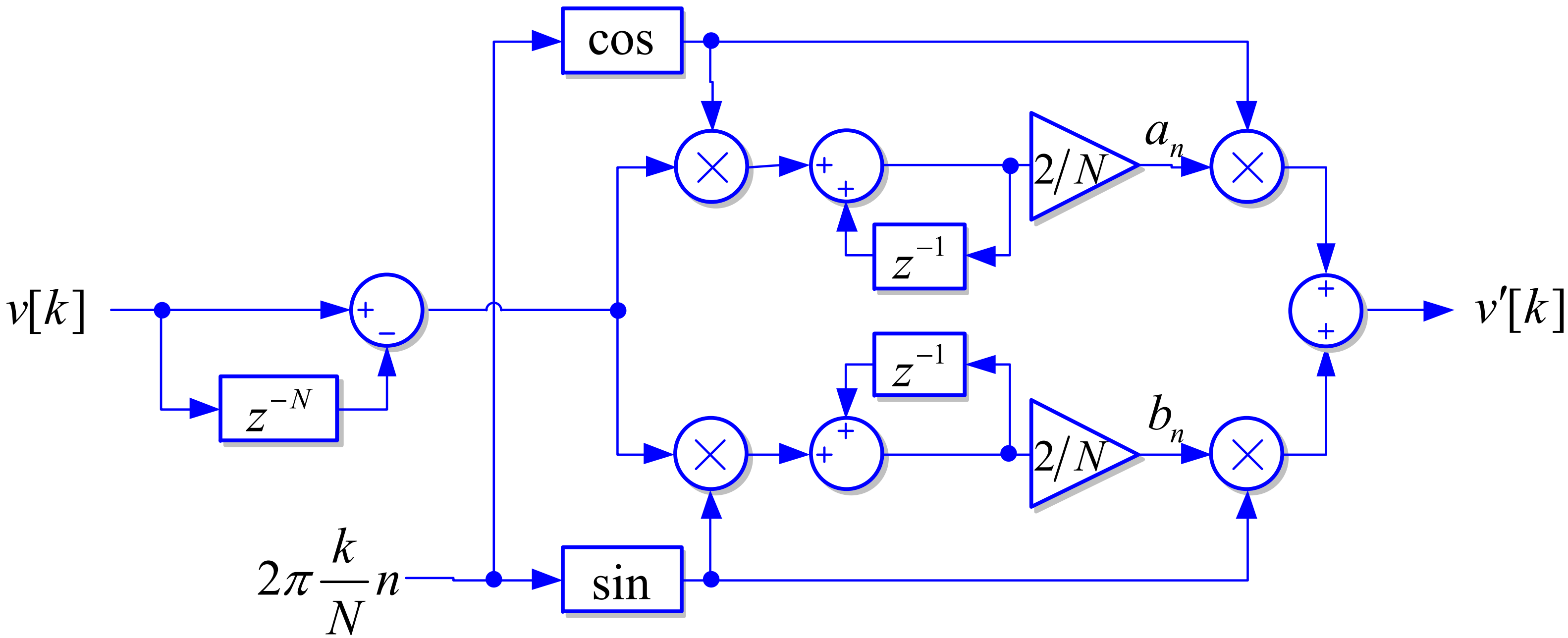
The amplitude of the correspondent n^{th} harmonic can be reconstructed in the time domain by:

$$v[k] = \frac{2}{N}V[n]\cdot e^{j2\pi\frac{n}{N}k}$$

Grid synchronization by using Fourier analysis

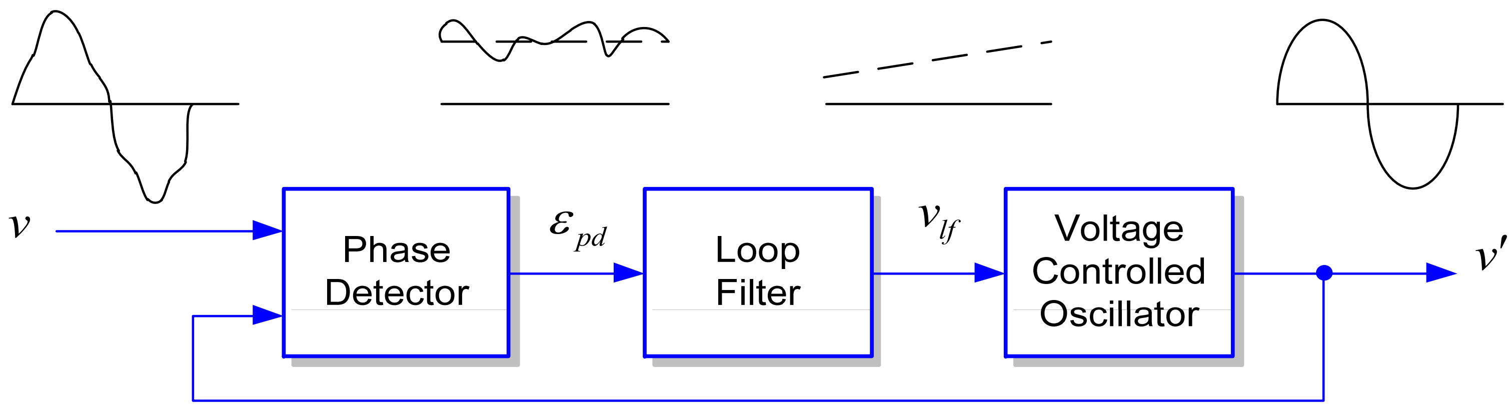
- Discrete adaptive band pass filter based on RDFT

  The RDFT can be applied to implement a discrete adaptive band pass filter to extract the n^{th} frequency component of the input signal.



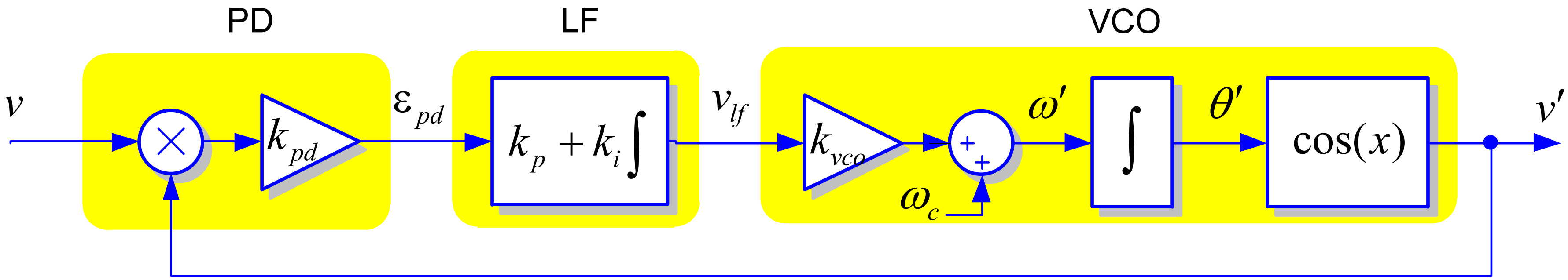
Grid synchronization by using a PLL

- Basic structure of a Phase-locked Loop (PLL)
  - Phase Detector (PD). This block generates an output signal proportional to the phase difference between v and v'. Depending on the type of PD, high frequency ac components appear together the dc phase difference signal
  - Loop Filter (LF). This block exhibits low pass characteristic and filters out the high frequency ac components from the PD output. Typically this is a 1st order LPF or PI controller
  - Voltage Controlled Oscillator (VCO). This block generates at its output an ac signal whose frequency varies respect a central frequency as a function of the input voltage



Grid synchronization by using a PLL

- PLL's basic equations



Input signal: $v = V\sin(\theta) = V\sin(\omega t + \phi)$

VCO output: $v' = \cos(\theta') = \cos(\omega' t + \phi')$

Multiplier PD output: $\varepsilon_{pd} = Vk_{pd}\sin(\omega t + \phi)\cos(\omega' t + \phi')$

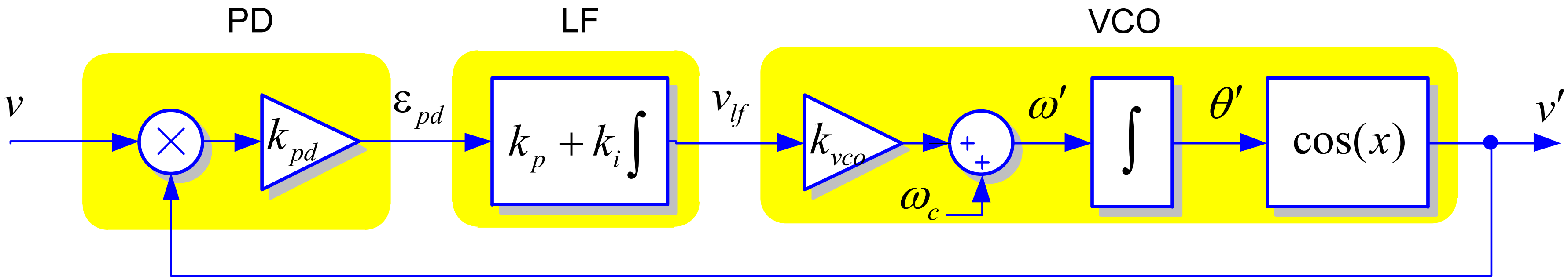
$$= \frac{Vk_{pd}}{2}\left[\underbrace{\sin\left((\omega-\omega')t + (\phi-\phi')\right)}_{low\text{-}frequency\ term} + \underbrace{\sin\left((\omega+\omega')t + (\phi+\phi')\right)}_{high\text{-}frequency\ term}\right]$$

The high-frequency components of the PD error signal will be cancelled out by the LF. Therefore, the PD error signal to be considered in this analysis is:

$$\bar{\varepsilon}_{pd} = \frac{Vk_{pd}}{2}\sin\left((\omega-\omega')t + (\phi-\phi')\right)$$

Grid synchronization by using a PLL

- PLL's basic equations



Let's asume the VCO is well tuned to the input frequency, that is $\omega \approx \omega'$

Therefore, the dc term of the phase error signal is given by:

$$\bar{\varepsilon}_{pd} = \frac{Vk_{pd}}{2}\sin(\phi - \phi')$$

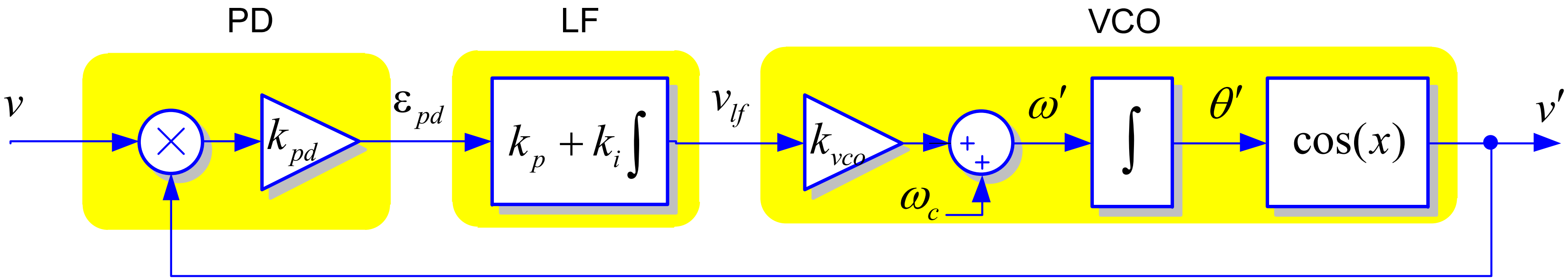
The multiplier PD produces nonlinear phase detection because of the sinusoidal function. However, when $\phi \approx \phi'$, the output of the multiplier PD can be linearized in the vicinity of such operating point since $\sin(\phi - \phi') \approx \sin(\theta - \theta') \approx (\theta - \theta')$.

The relevant term of the phase error signal when the PLL is locked is given by:

$$\bar{\varepsilon}_{pd} = \frac{Vk_{pd}}{2}(\theta - \theta')$$

Grid synchronization by using a PLL

- PLL's basic equations



The averaged frequency of the VCO is determined by:

$$\overline{\omega}' = \left(\omega_c + \Delta\overline{\omega}'\right) = \left(\omega_c + k_{vco}\overline{v}_{lf}\right)$$

ω_c is the center frequency of the VCO and it is supplied to the PLL as a feed-forward parameter dependent on the range of frequency to be detected.

Small signal variations in the VCO frequency are given by:

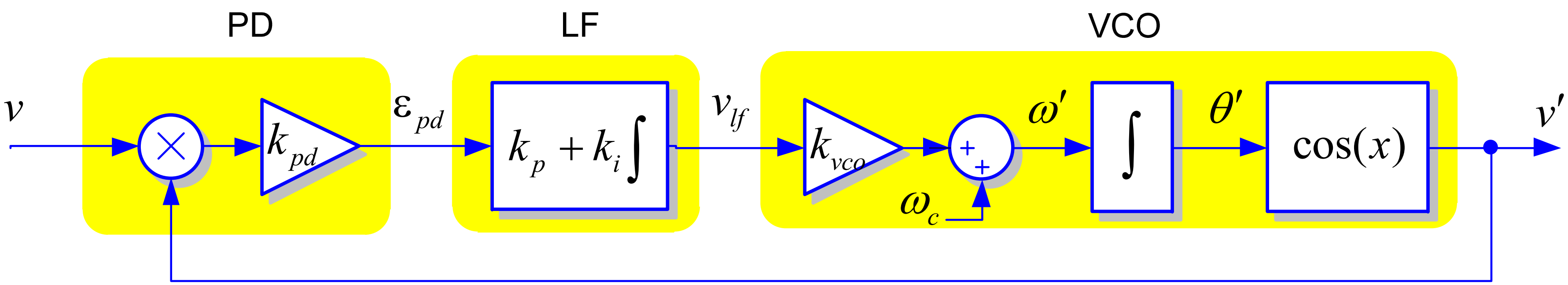
$$\tilde{\omega}' = k_{vco}\tilde{v}_{lf}$$

Variations in the phase-angle detected by the PLL can be written as:

$$\tilde{\theta}'(t) = \int \tilde{\omega}' dt = \int k_{vco}\tilde{v}_{lf} dt$$

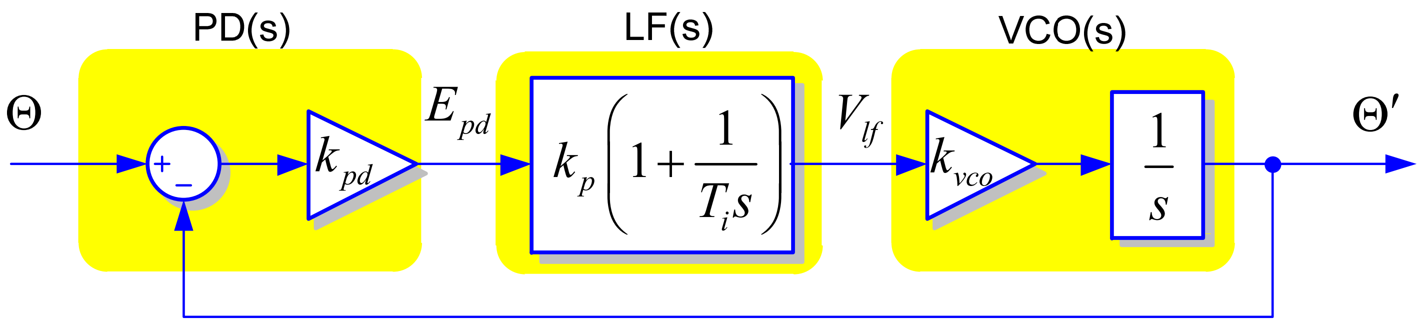
Grid synchronization by using a PLL

- Linearized small signal model of a PLL



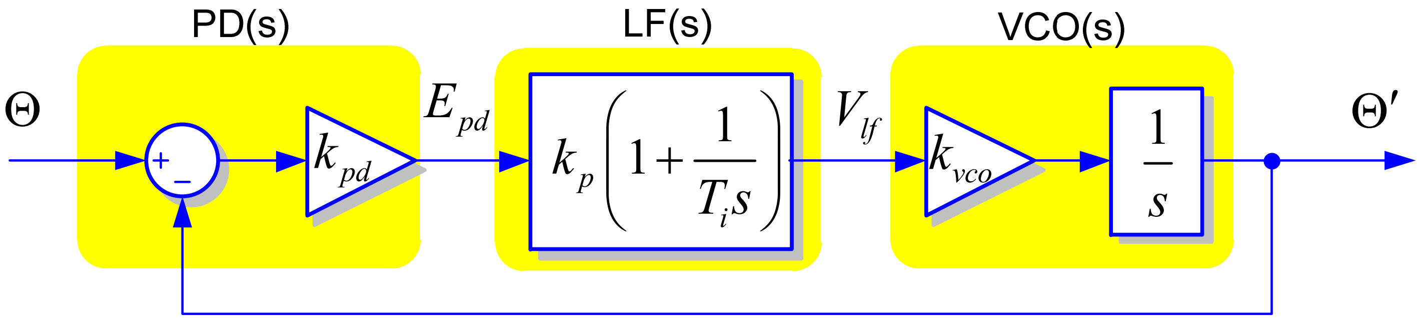
Expressions in the complex frequency domain by using the Laplace transform ($k_{pd}=k_{vco}=1$, $V=1$):

$$E_{pd}(s)=\frac{V}{2}\left(\Theta(s)-\Theta'(s)\right) \qquad V_{lf}(s)=k_p\left(1+\frac{1}{T_i s}\right)\varepsilon_{pd}(s) \qquad \Theta'(s)=\frac{1}{s}V_{lf}(s)$$



Grid synchronization by using a PLL

- Linearized small signal model of a PLL



Transfer functions:

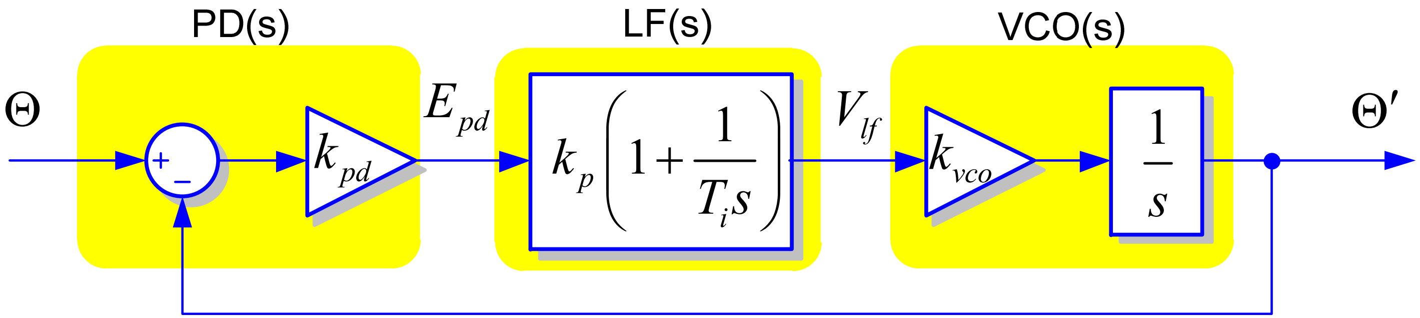
Open-loop phase transfer function $$F_{OL}(s) = PD(s)\cdot LF(s)\cdot VCO(s) = \frac{k_p s + \frac{k_p}{T_i}}{s^2}$$

Closed-loop phase transfer function $$H_\theta(s) = \frac{\Theta'(s)}{\Theta(s)} = \frac{K_p s + \frac{K_p}{T_i}}{s^2 + K_p s + \frac{K_p}{T_i}}$$

Closed-loop error phase transfer function $$E_\theta(s) = \frac{E_{pd}(s)}{\Theta(s)} = \frac{s^2}{s^2 + K_p s + \frac{K_p}{T_i}}$$

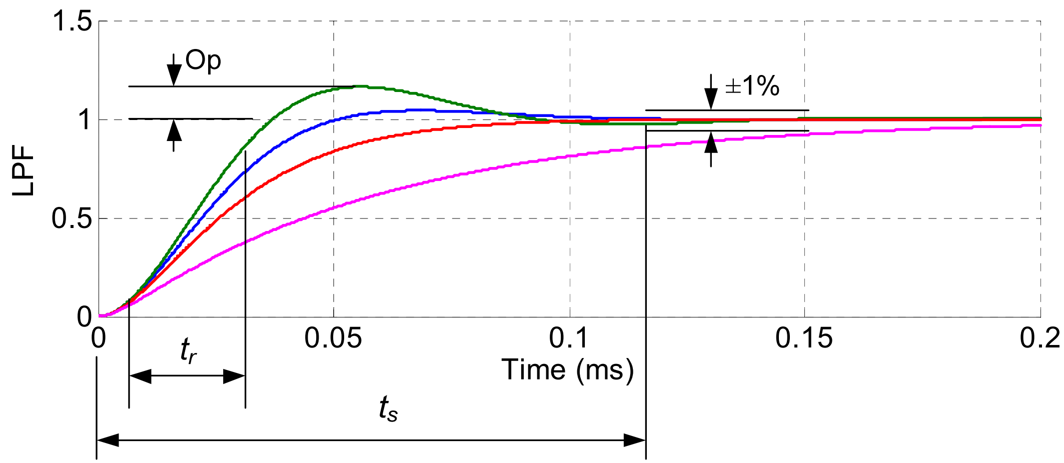
Grid synchronization by using a PLL

- Linearized small signal model of a PLL



General form of second order transfer functions:

$$H_\theta(s)=\frac{2\zeta\omega_n s+\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2}\qquad E_\theta(s)=\frac{s^2}{s^2+2\zeta\omega_n s+\omega_n^2}\qquad \omega_n=\sqrt{\frac{K_p}{T_i}}\text{ and }\xi=\frac{\sqrt{K_pT_i}}{2}$$



Settling time:

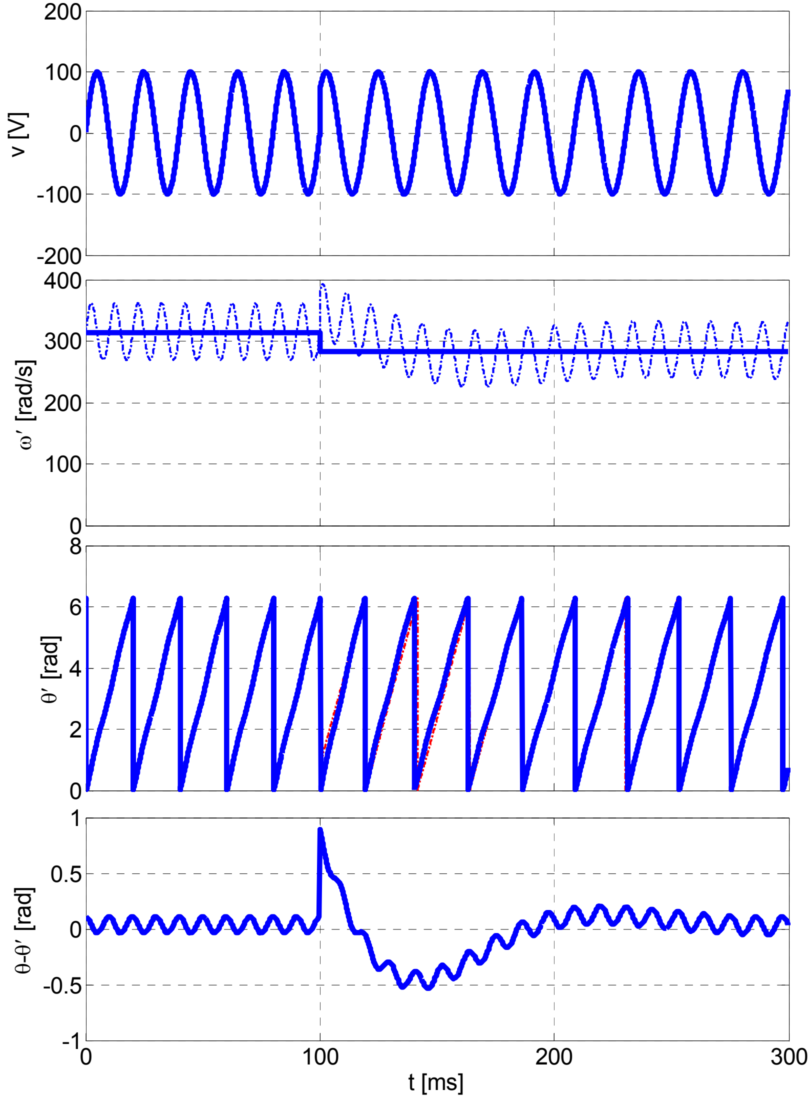
$$t_S=4.6\cdot\tau\,;\quad \tau=\frac{1}{\xi\omega_n}$$

with $\xi=\frac{1}{\sqrt{2}}$:

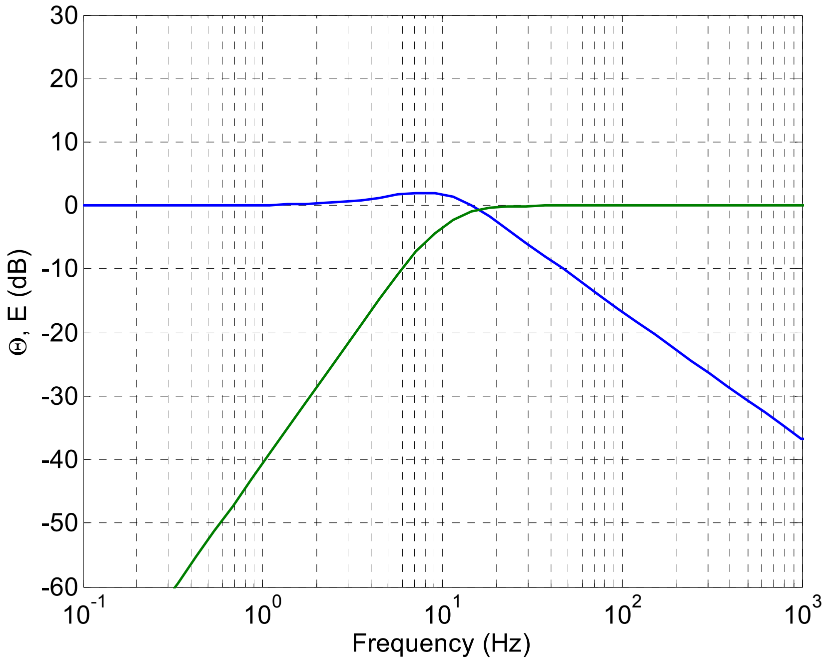
$$K_p=2\xi\omega_n=\frac{9.2}{t_s}\,,\quad T_i=\frac{2\xi}{\omega_n}=\frac{t_s\xi^2}{2.3}$$

Grid synchronization by using a PLL

- PLL response



$$\xi = \frac{1}{\sqrt{2}} \quad ; \quad t_s = 100ms \;\; ; \;\; \omega_n = 10.34Hz$$



$$\omega_{-3dB} = \omega_n \left[1 + 2\xi^2 + \sqrt{\left(1+2\xi\right)^2 + 1} \right]^{1/2}$$

$$\omega_{-3dB} = 2.06 \cdot \omega_n$$

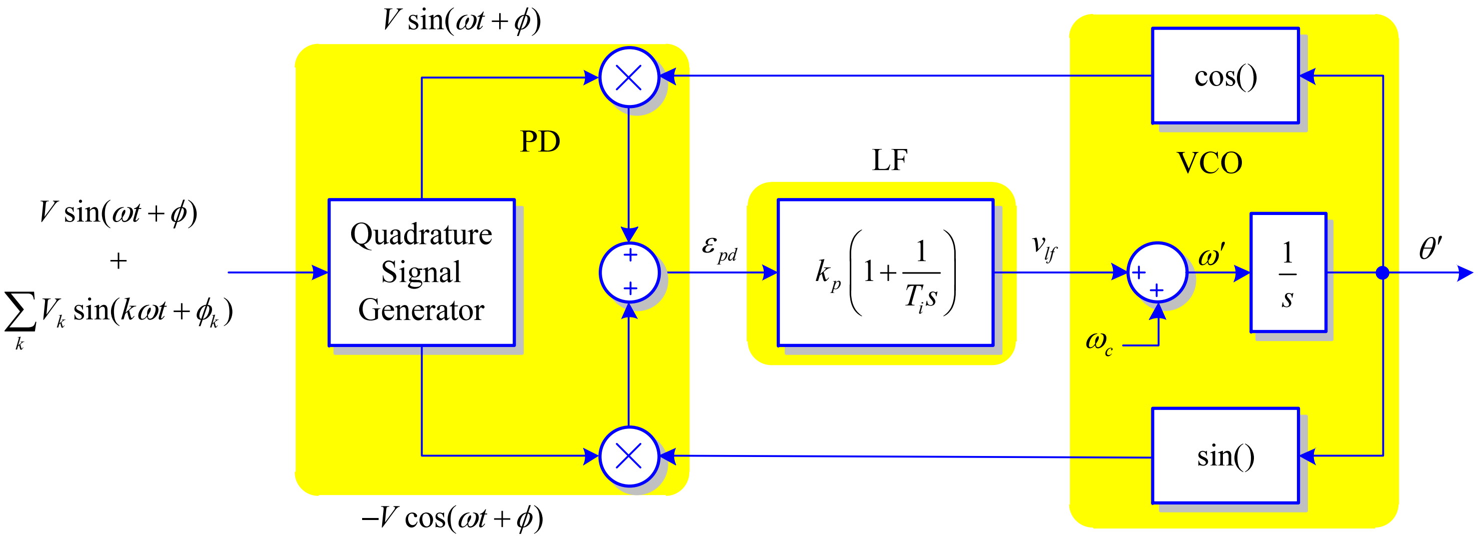
Phase detection based on in-quadrature signals

- Steady-state error cancellation in phase detection
  - Form an input signal such as $v = V\sin(\theta) = V\sin(\omega t+\phi)$ a Quadrature Signal Generator (QSG) is able to generates the following set of in-quadrature signals:

$$\mathbf{v}_{(\alpha\beta)} = \begin{bmatrix} v_\alpha \\ v_\beta \end{bmatrix} = V\begin{bmatrix} \sin(\omega t+\phi) \\ -\cos(\omega t+\phi) \end{bmatrix} = V\begin{bmatrix} \sin(\theta) \\ -\cos(\theta) \end{bmatrix}$$

  - A phase detector (PD) based on a QSG can cancel-out the oscillations at twice the input frequency in the detected phase-angle error signal of the multiplier PD, since:

$$\begin{aligned} \varepsilon_{pd} &= V\sin(\omega t+\phi)\cos(\omega' t+\phi') - V\cos(\omega t+\phi)\sin(\omega' t+\phi') \\ &= V\sin\big((\omega-\omega')t + (\phi-\phi')\big) = V\sin(\theta-\theta') \end{aligned}$$



Phase detection based on in-quadrature signals

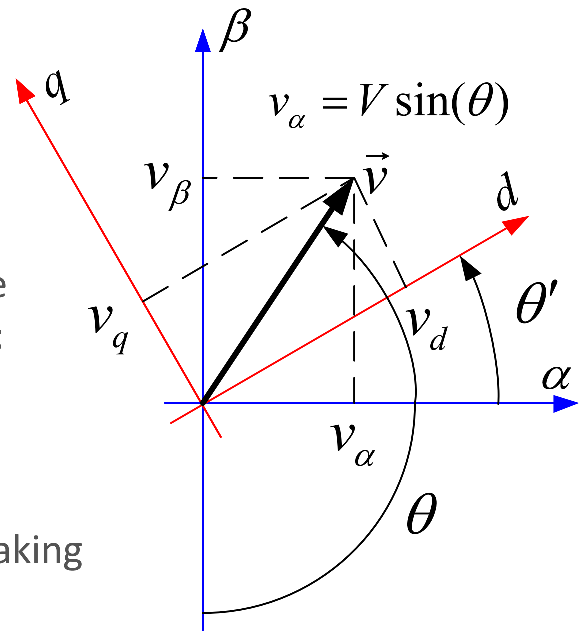
- Park transformation applied to phase detection
  - The in-quadrature signals (v_α, v_β) define a voltage vector **v**:

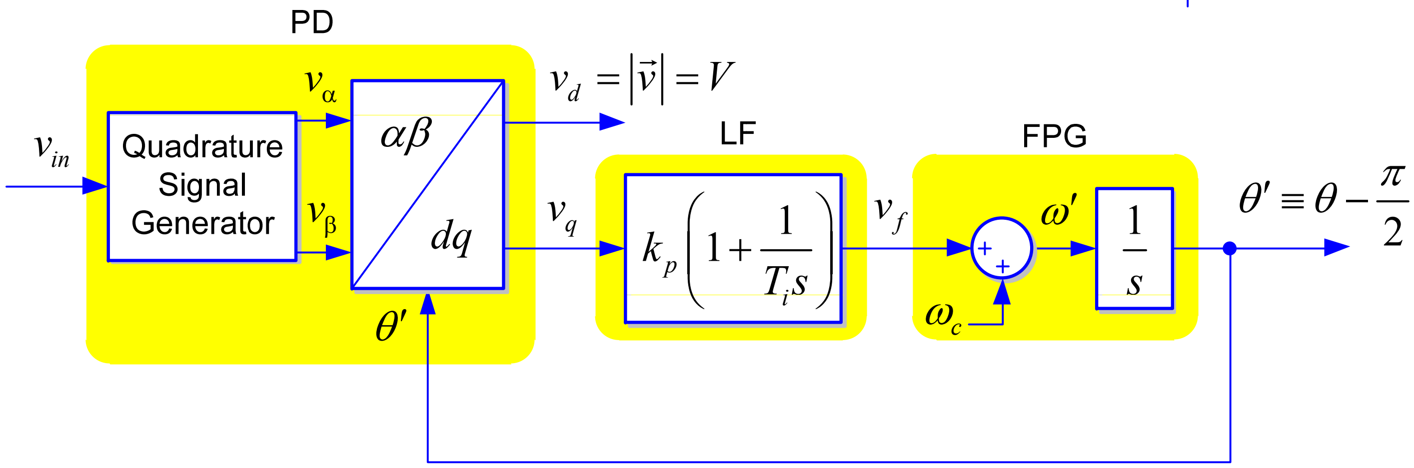
$$\mathbf{v}_{(\alpha\beta)} = \begin{bmatrix} v_\alpha \\ v_\beta \end{bmatrix} = V \begin{bmatrix} \sin(\theta) \\ -\cos(\theta) \end{bmatrix}$$

  - The Park transformation gives allows expressing the voltage vector **v** on a d-q synchronous reference frame (SRF) set at θ':

$$\mathbf{v}_{(dq)} = \begin{bmatrix} v_d \\ v_q \end{bmatrix} = \begin{bmatrix} \cos(\theta') & \sin(\theta') \\ -\sin(\theta') & \cos(\theta') \end{bmatrix} \begin{bmatrix} v_\alpha \\ v_\beta \end{bmatrix} = V \begin{bmatrix} \sin(\theta - \theta') \\ -\cos(\theta - \theta') \end{bmatrix}$$

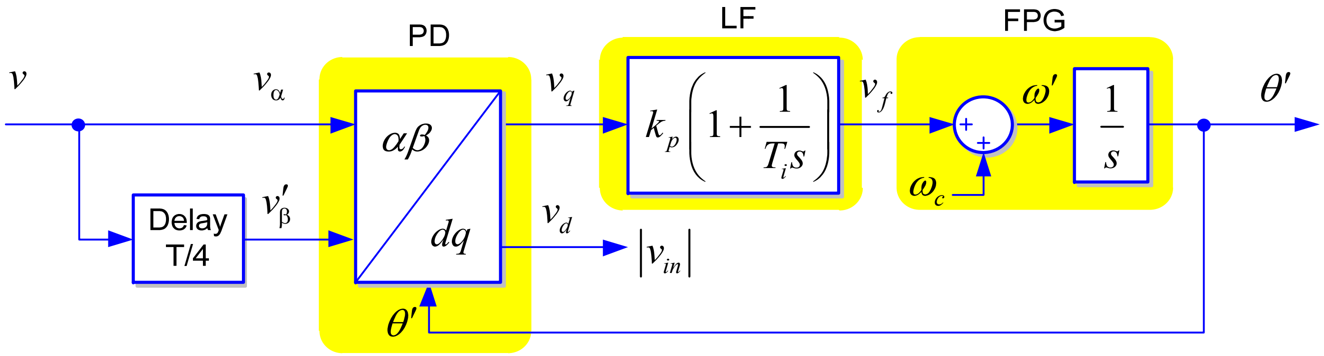
  - The LF of the PLL set the angular position of the SRF by making $v_q=0$ and achieving $\theta'=\theta$





PLL based on a T/4 transport delay

- The transport delay block of can be effortlessly is easily programmed through the use of a first-in-first-out (FIFO) buffer, whose size is set to one fourth of the number of samples contained in one cycle of the fundamental frequency
- If the grid voltage frequency changes in respect its rated value, the output signals of the QSG will not be perfectly orthogonal, which will give rise to errors in the PLL synchronization
- If input voltage consists of several frequency components, orthogonal signals generation will produce errors because each of the components should be delayed one fourth of its fundamental period



PLL based on the Hilbert transform

- The time-domain Hilbert transform: $\hat{g}(t) = \dfrac{1}{\pi}\displaystyle\int_{-\infty}^{\infty} \dfrac{g(\tau)}{t-\tau} d\tau \quad \Leftrightarrow \quad \hat{g}(t) = \dfrac{1}{\pi t} * g(t)$

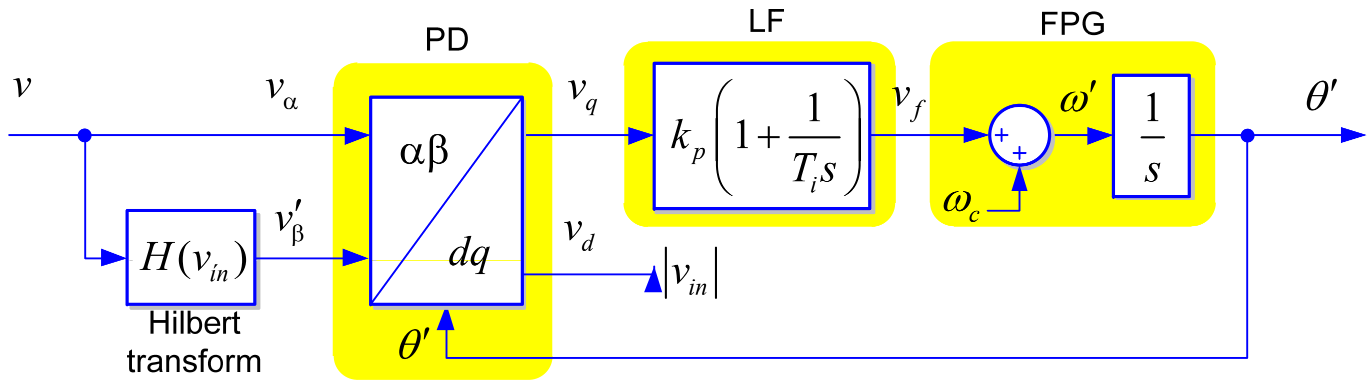
$$hilbert\left(e^{jkt}\right) = \frac{1}{\pi}\int_{-\infty}^{\infty} \frac{e^{jk\tau}}{t-\tau} d\tau = -je^{jkt}\Big|_{k>0}$$

$$hilbert\left(\sin(kt)\right) = hilbert\left(\frac{e^{jkt} - e^{-jkt}}{2j}\right) = -\frac{e^{jkt} + e^{-jkt}}{2} = -\cos(kt)$$

$$\boxed{\sin(kt) \rightarrow -\cos(kt) \rightarrow -\sin(kt) \rightarrow \cos(kt) \rightarrow \sin(kt)}$$

- Hilbert transform is also called a “quadrature filter”.

Fourier transform: $F\left(\dfrac{1}{\pi t}\right) = -j\ sign(f) \begin{cases} -j & for \quad f > 0 \\ 0 & for \quad f = 0 \\ +j & for \quad f < 0 \end{cases}$

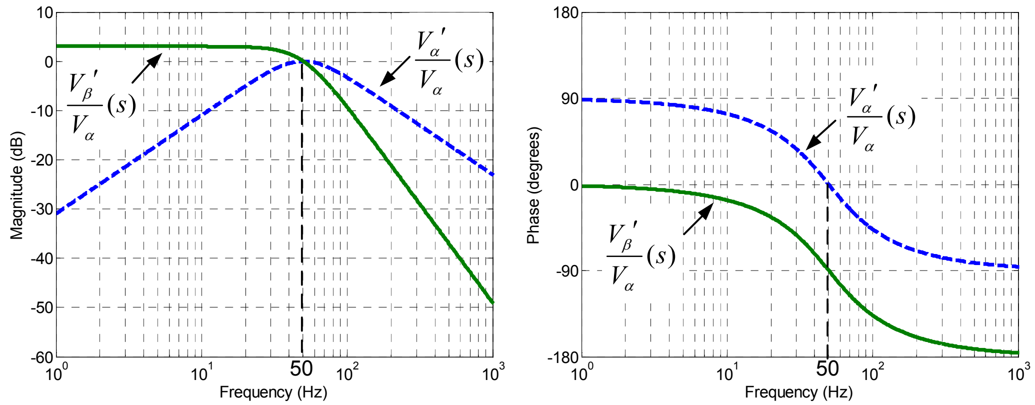


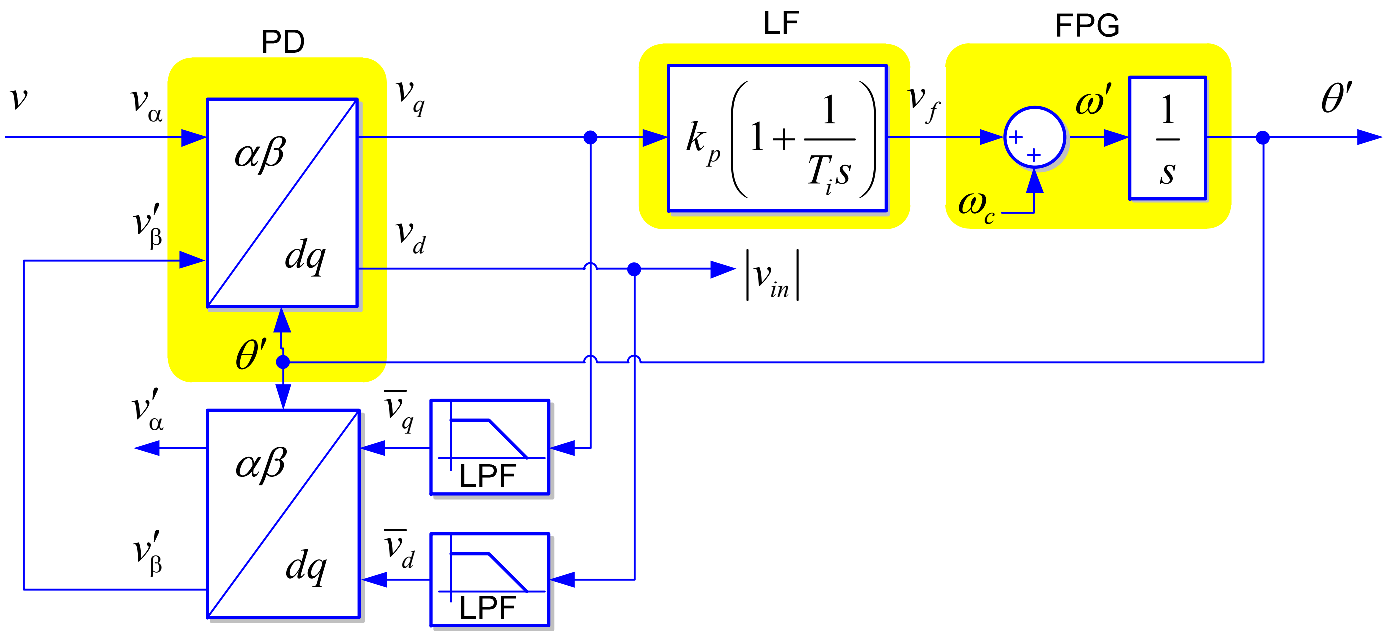
PLL based on the inverse Park transform

- The invers Park transform technique allows QSG plus noise filtering

$$\frac{V_\beta'}{V_\alpha}(s) = \frac{k\omega'^2}{s^2 + sk\omega' + \omega'^2} \quad ; \quad k = \frac{\omega_f}{\omega'}$$

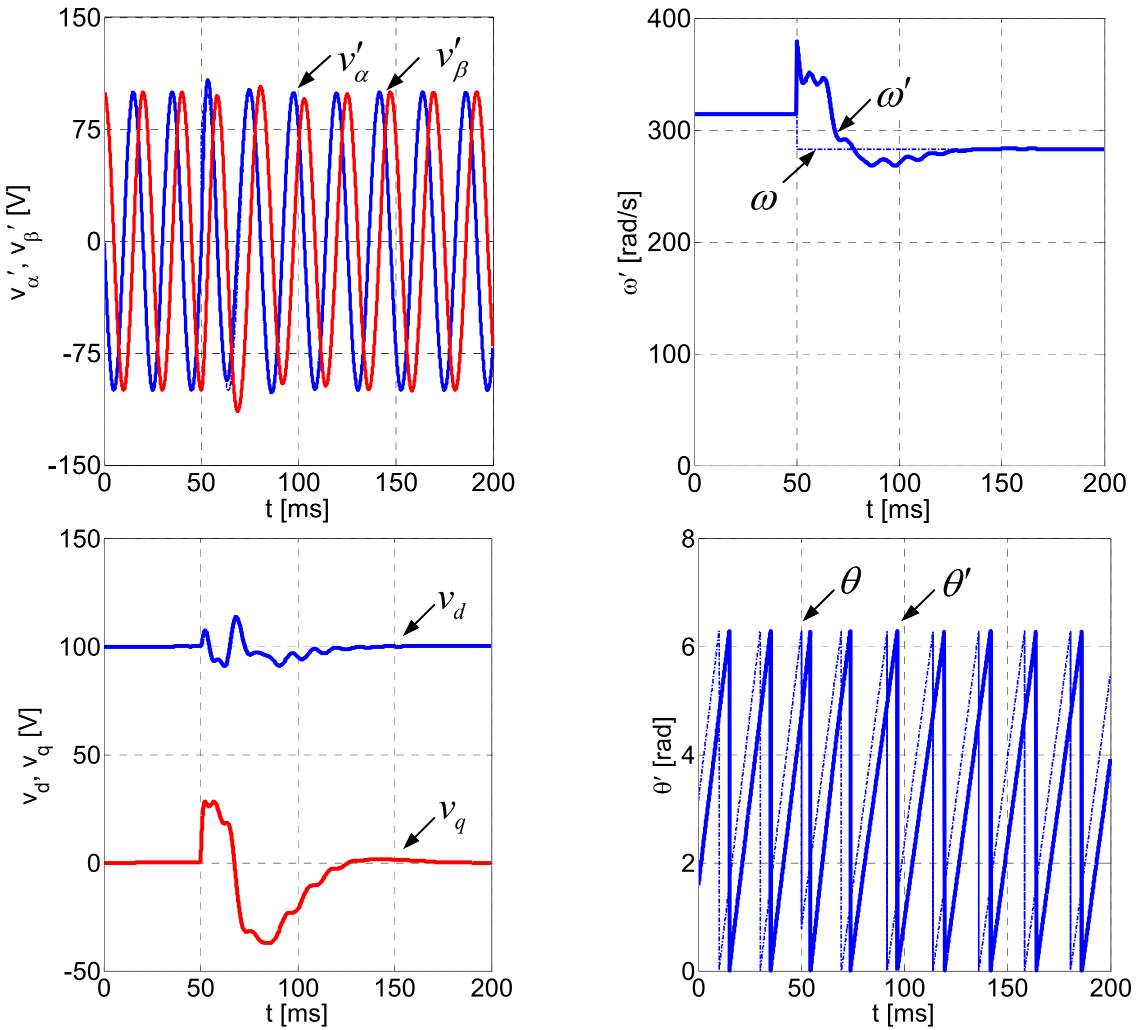
$$\frac{V_\alpha'}{V_\alpha}(s) = \frac{sk\omega'}{s^2 + sk\omega' + \omega'^2} \quad ; \quad k = \frac{\omega_f}{\omega'}$$





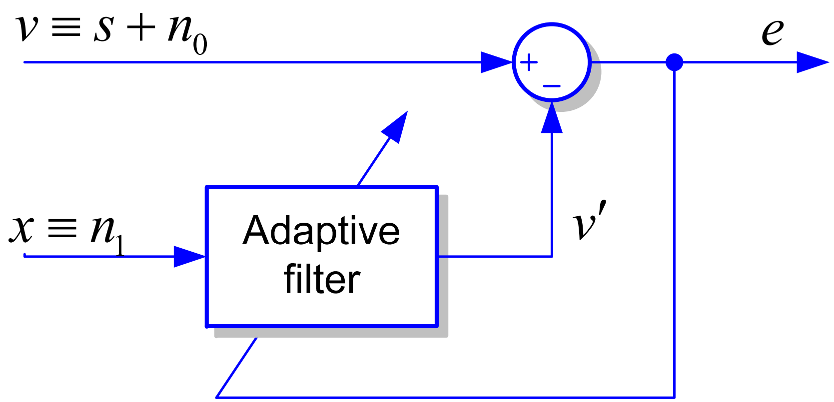
PLL based on the inverse Park transform

- Response of the inverse Park transform PLL in the presence of a phase (+45º) and frequency (50 to 45 Hz) jump in the input signal



Adaptive filtering

- Adaptive Noise Cancellation
  - An adaptive filter is a filter that has the ability of adjust its own parameters automatically according to an optimization algorithm
  - Adaptive Noise Cancelling (ANC) is an application of adaptive filtering, in which an auxiliary reference signal n_1, correlated to the primary noise signal n_0, is adaptively filtered to produce an output signal that is as close a replica as possible of n_0. This output signal is subtracted from the primary input. As a result, the primary noise n_0 is eliminated by cancellation
  - When the ANC technique is used to cancel out specific frequency components of the input signal, this filtering concept is also called Adaptive Notch Filtering (ANF)

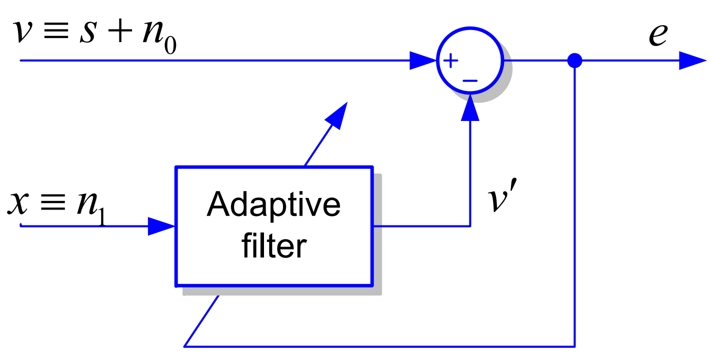


ANC based on the LMS algorithm

The most extended adaptation algorithm used to set the weights of the adaptive filter is the least-mean-squares (LMS) algorithm.

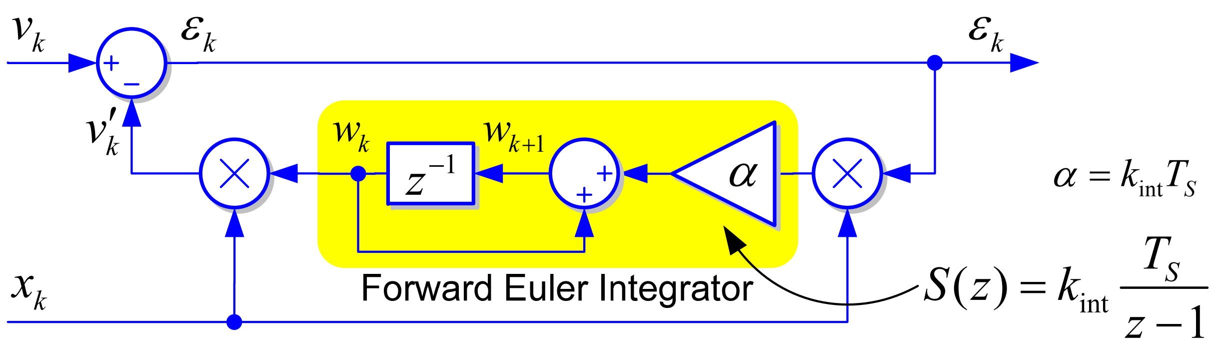
- Reference signal vector: $\mathbf{x}_k = [x_k, x_{k-1}, \ldots, x_{k-N}]$
- Weights vector: $\mathbf{w}_k = [w_k, w_{k-1}, \ldots, w_{k-N}]$
- LMS algorithm:

$$v'_k = \mathbf{w}_k^T \cdot \mathbf{x}_k \; ;$$
$$e_k = v_k - v'_k \; ;$$
$$\mathbf{w}_{k+1} = \mathbf{w}_k + \alpha e_k \mathbf{x}_k$$



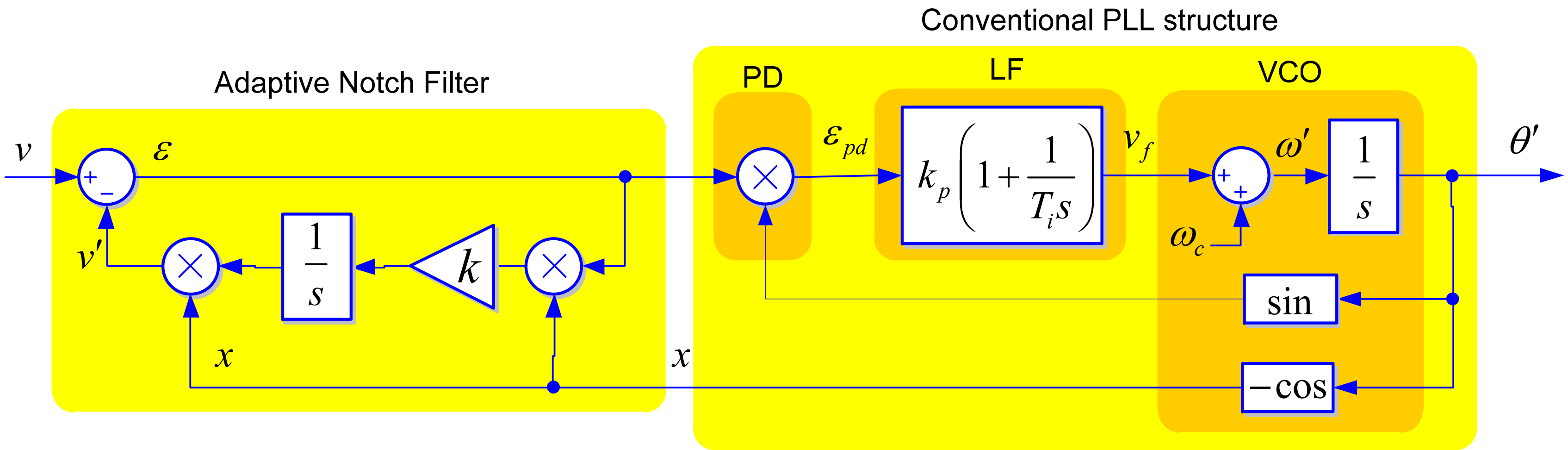
- Estimate of the negative gradient: $e_k \cdot \mathbf{x}_k$
- Adaptation gain: α

Schematic representation of a very simple LMS algorithm with only one weight:

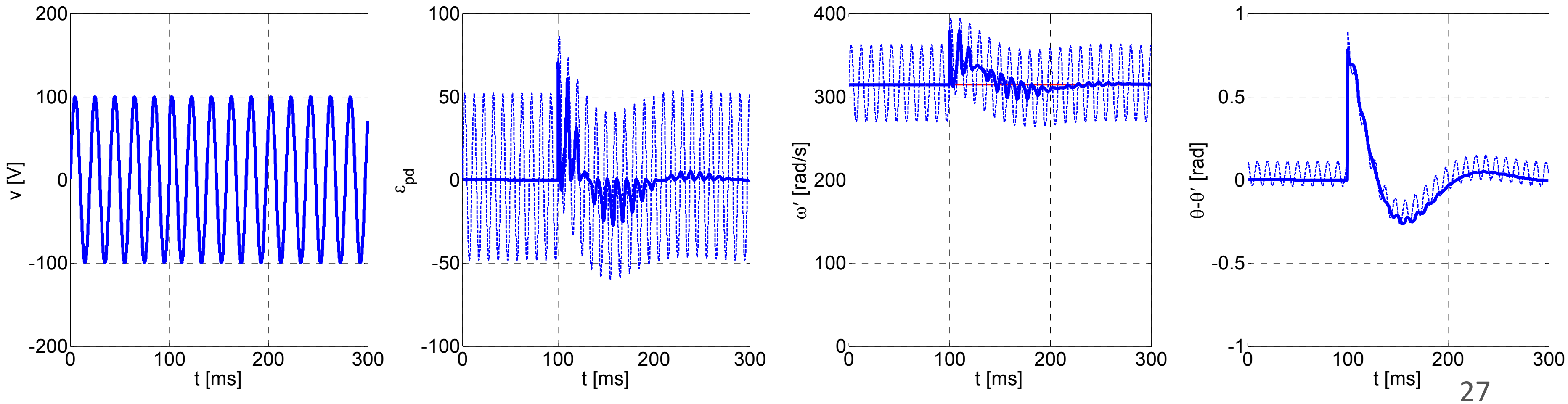


The Enhanced PLL

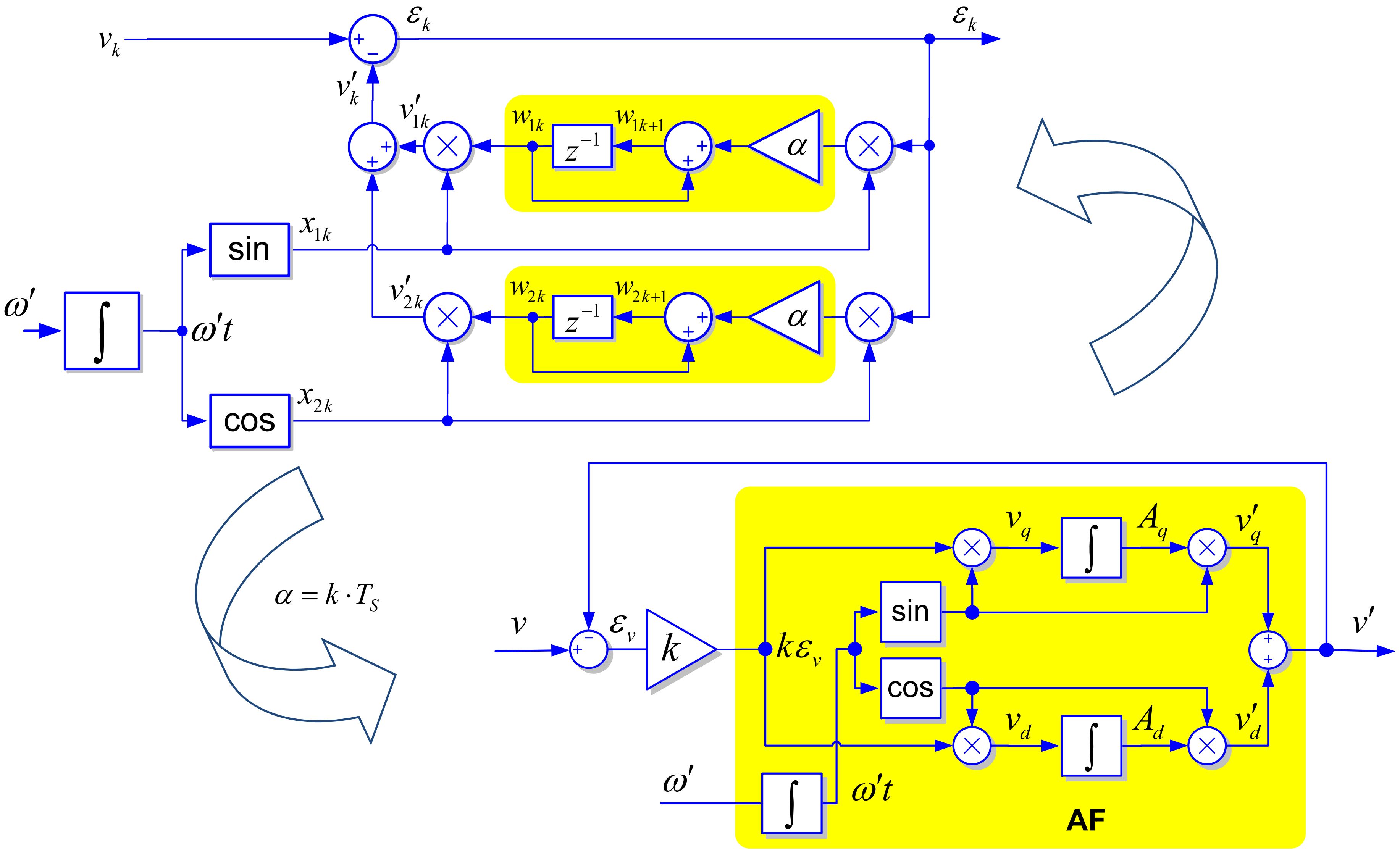
- Adaptive Notch Filter + Conventional PLL -> Enhanced PLL



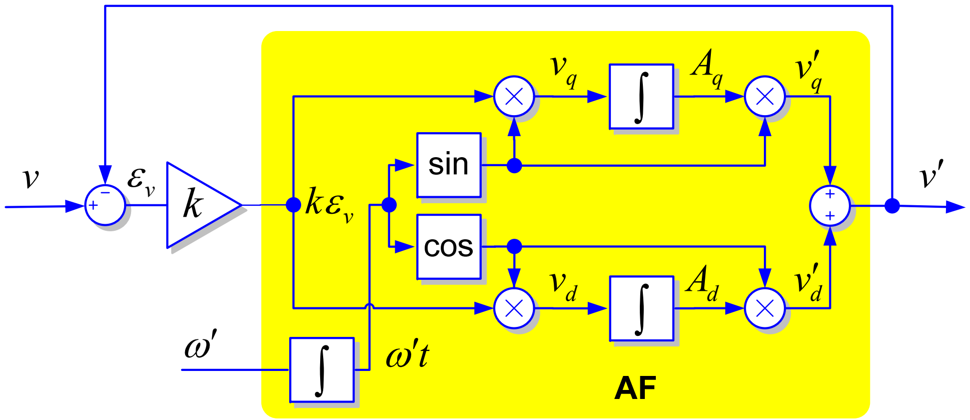
EPLL response:



Adaptive Notch Filter with Two Weights



Adaptive Notch Filter



Transfer function of the AF:

$$\boxed{AF(s) = \frac{v'}{k\varepsilon_v}(s) = \frac{s}{s^2 + \omega'^2}}$$

The AF as a resonator with infinite gain for any sinusoid with frequency ω' applied to its input.

Development:

Defining $g = k\varepsilon_v$

$$v_d = g\cos(\omega' t) = \frac{1}{2} g\left[e^{j\omega' t} + e^{-j\omega' t}\right] \qquad v_q = g\sin(\omega' t) = \frac{1}{j2} g\left[e^{j\omega' t} - e^{-j\omega' t}\right]$$

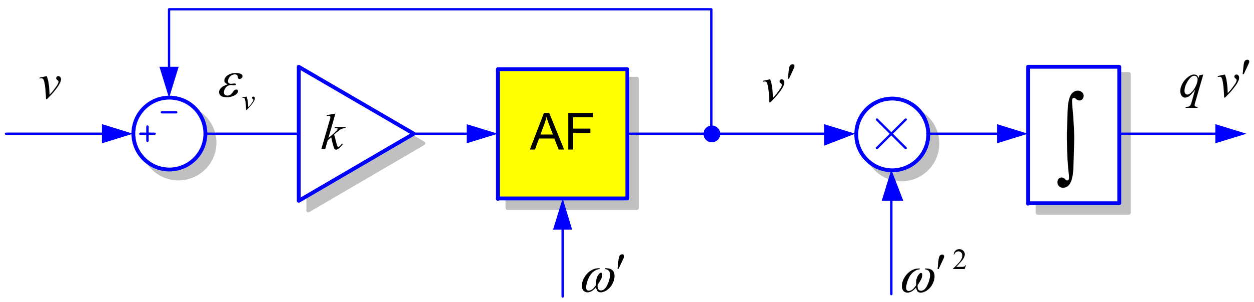
$$A_d(s) = \frac{1}{s} v_d(s) = \frac{1}{2s}\left[g(s + j\omega' t) + g(s - j\omega' t)\right] \qquad A_q(s) = \frac{1}{s} v_q(s) = \frac{1}{j2s}\left[g(s + j\omega' t) - g(s - j\omega' t)\right]$$

$$v'_d(s) = \frac{1}{2}\left[A_d(s + j\omega' t) + A_d(s - j\omega' t)\right] = \frac{1}{4(s + j\omega')}\left[g(s) + g(s + 2j\omega')\right] + \frac{1}{4(s - j\omega')}\left[g(s) + g(s - 2j\omega')\right],$$

$$v'_q(s) = \frac{1}{2j}\left[A_q(s + j\omega' t) - A_q(s - j\omega' t)\right] = \frac{1}{4(s + j\omega')}\left[g(s) - g(s + 2j\omega')\right] + \frac{1}{4(s - j\omega')}\left[g(s) - g(s - 2j\omega')\right].$$

$$v'(s) = v'_d(s) + v'_q(s) = \frac{s}{s^2 + \omega'^2} g(s)$$

Adaptive Notch Filter



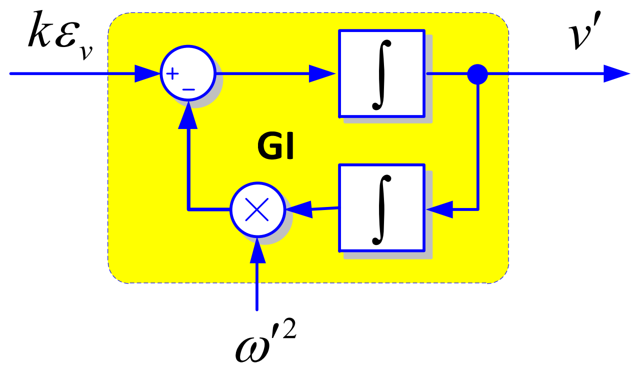
Transfer functions

- Adaptive filter $$AF(s) = \frac{v'}{k\varepsilon_v}(s) = \frac{s}{s^2 + \omega'^2}$$
- Adaptive bandpass filter $$ABPF(s) = \frac{v'}{v}(s) = \frac{AF(s)}{1 + AF(s)} = \frac{ks}{s^2 + ks + \omega'^2}$$
- Adpative nothch filter $$ANF(s) = \frac{\varepsilon_v}{v}(s) = 1 - ABPF(s) = \frac{s^2 + \omega'^2}{s^2 + ks + \omega'^2}$$
- Adaptive low/pass filter $$ALPF(s) = \frac{qv'}{v}(s) = \frac{\omega'^2}{s} ABPF(s) = \frac{k\omega'^2}{s^2 + ks + \omega'^2}$$

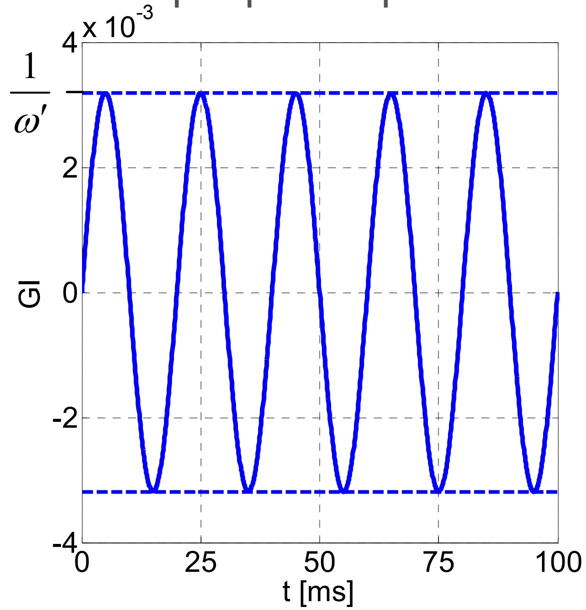
Generalized Integrator (GI)

- The adaptive filter is also referred as a 'sinusoidal integrator' or 'generalized integrator' (GI) because of its response to sinusoidal input signals

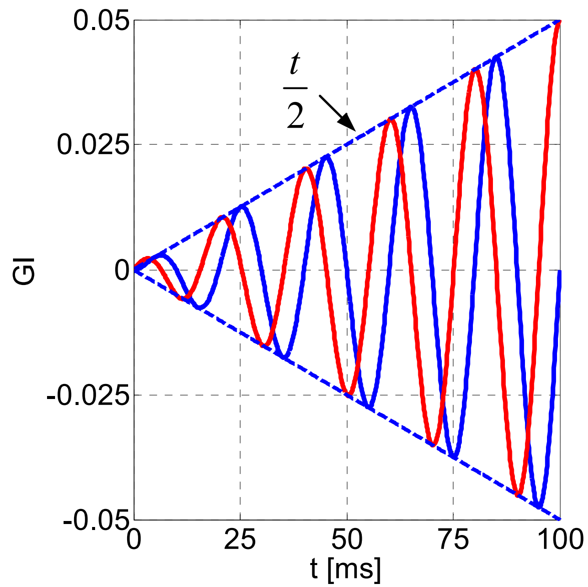
$$GI(s) = \frac{v'}{k\varepsilon_v}(s) = \frac{s}{s^2 + \omega'^2}$$



Step input response

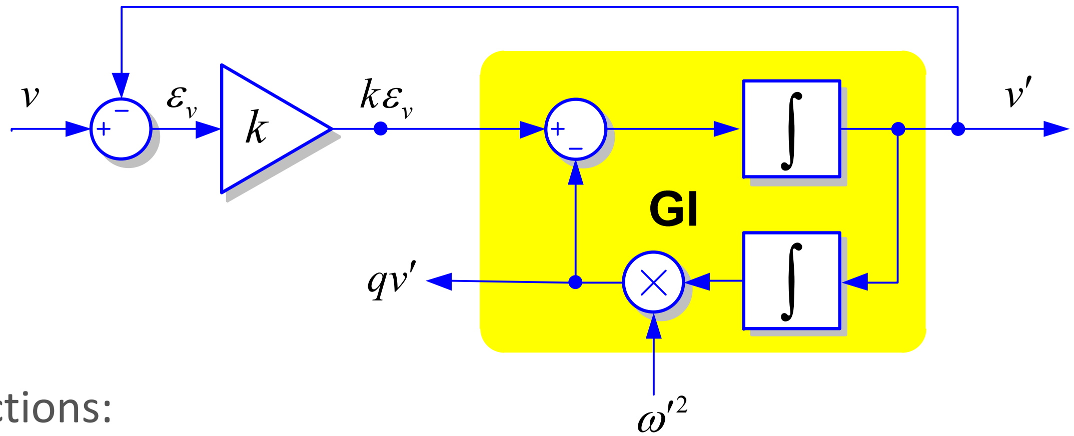


Sinusoidal input response



2^{nd}-order AF on the GI

- Adaptive filter:



- Transfer functions:

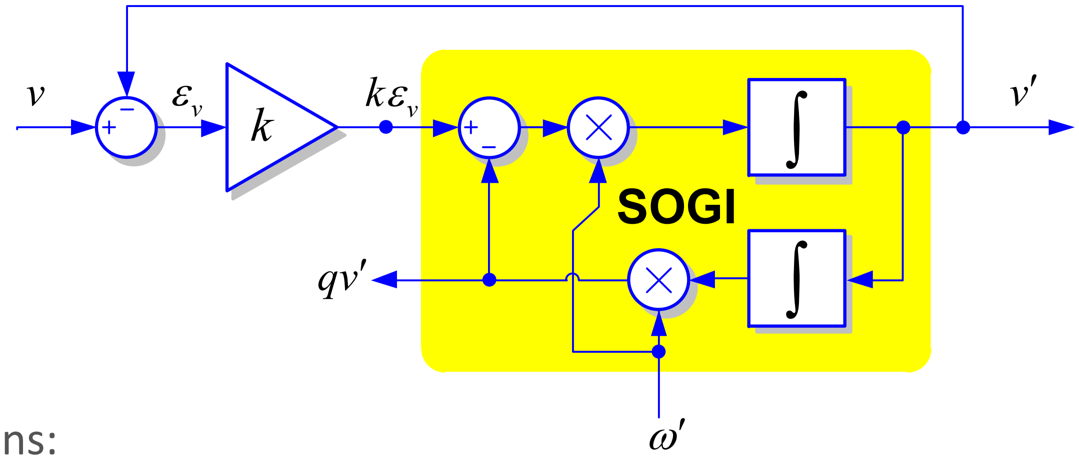
$$D(s) = \frac{v'}{v}(s) = \frac{ks}{s^2 + ks + \omega'^2} \qquad Q(s) = \frac{qv'}{v}(s) = \frac{k\omega'^2}{s^2 + ks + \omega'^2}$$

v' and qv' signals are in-quadrature, which makes it suitable to implement a PLL based on QSG. The bandwidth and the static gain of $D(s)$ and $Q(s)$, respectively, are not only a function of the gain k, but they also depend on the center frequency of the filter, ω'. This issue can become an inconvenience when designing variable-frequency systems, as is the case of a PLL

Second Order Generalized Integrator (SOGI)

- The Second Order Generalized Integrator (SOGI) is an alternative sinusoidal integrator with the following transfer function:

$$SOGI(s) = \frac{v'}{k\varepsilon_v}(s) = \frac{\omega' s}{s^2 + \omega'^2}$$



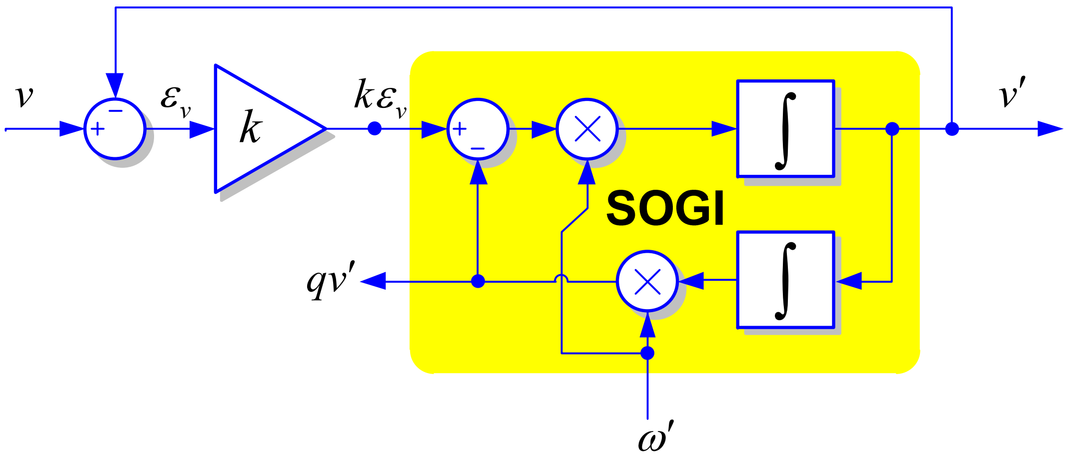
- Transfer functions:

$$D(s) = \frac{v'}{v}(s) = \frac{k\omega' s}{s^2 + k\omega' s + \omega'^2} \qquad Q(s) = \frac{qv'}{v}(s) = \frac{k\omega'^2}{s^2 + k\omega' s + \omega'^2}$$

The bandwidth and static gain of these transfer functions only depend on the value of k

Second Order Generalized Integrator (SOGI)

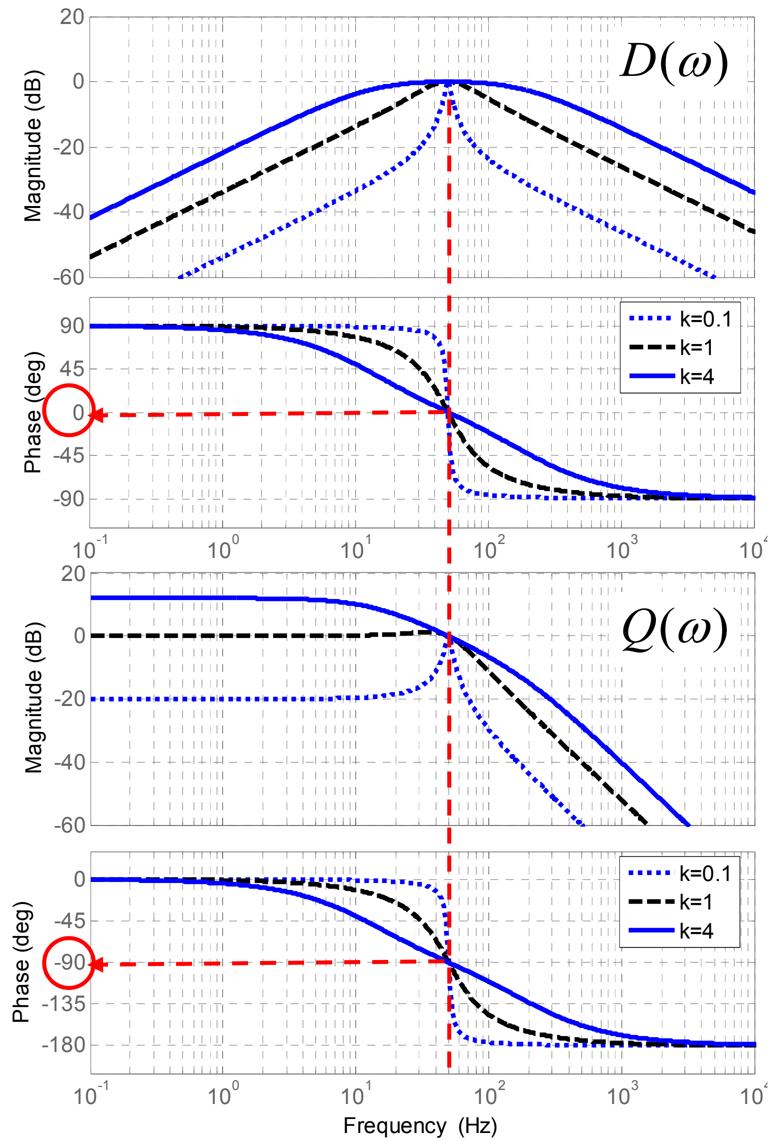
- Quadrature Signal Generation based on the SOGI



- Transfer functions:

$$D(s) = \frac{v'}{v}(s) = \frac{k\omega' s}{s^2 + k\omega' s + \omega'^2}$$

$$Q(s) = \frac{qv'}{v}(s) = \frac{k\omega'^2}{s^2 + k\omega' s + \omega'^2}$$



Second Order Generalized Integrator (SOGI)

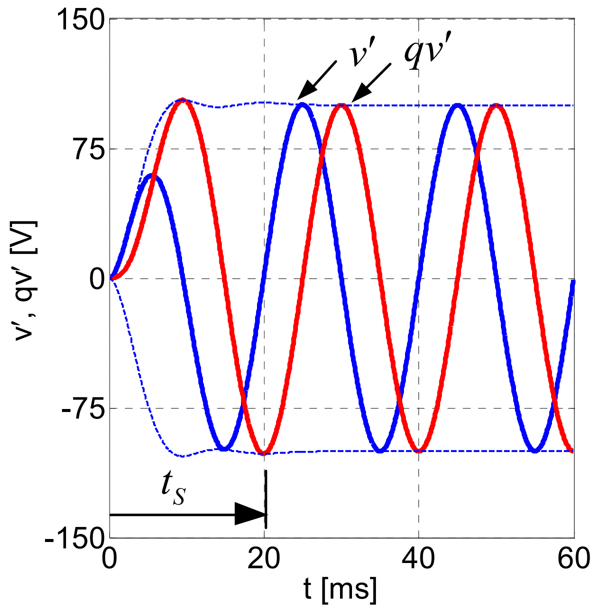
- Sinusoidal input response:

$$v' = -\frac{V}{\sqrt{1-(k/2)^2}} \sin\left(\omega\sqrt{1-(k/2)^2}\, t\right) e^{-\frac{k\omega}{2}t} + V \sin(\omega t)$$

$$qv' = \frac{V}{\sqrt{1-(k/2)^2}} \cos\left(\omega\sqrt{1-(k/2)^2}\, t - \varphi\right) e^{-\frac{k\omega}{2}t} - V \cos(\omega t)$$

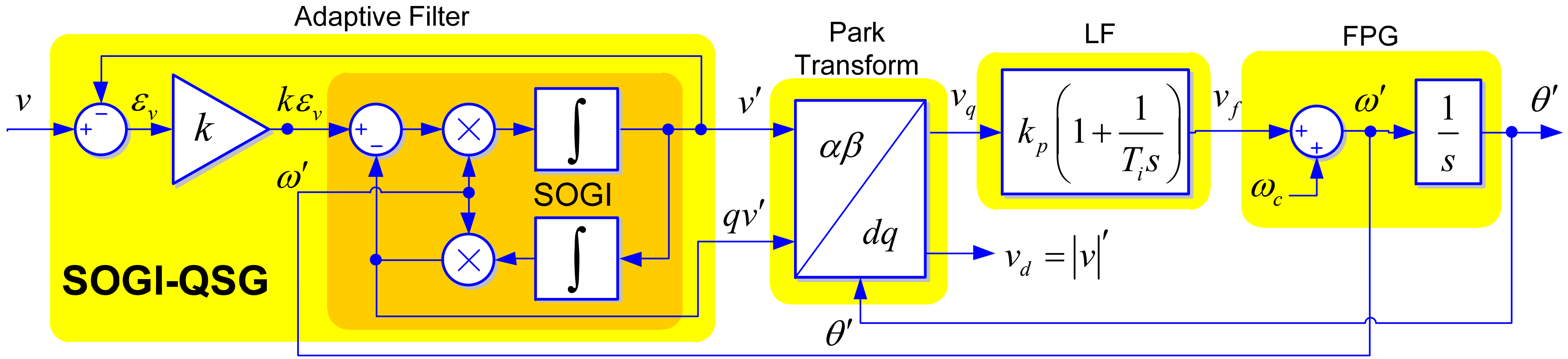
$$\varphi = \arctan\frac{k/2}{\sqrt{1-(k/2)^2}}$$

- Settle time: $\tau = 2/k\omega'$
- Constant time: $t_S = 4.6 \cdot \tau$
- Gain tuning: $k = \dfrac{9.2}{t_S \omega'}$

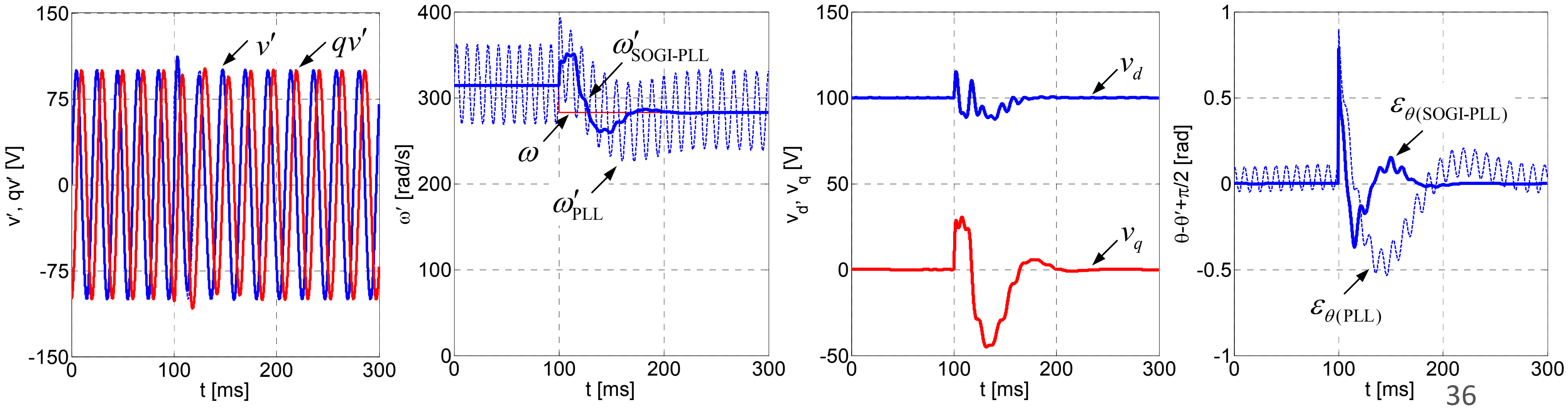


SOGI-PLL

- SOGI-QSG + Park-based PD + PLL



SOGI-PLL response:



SOGI-Frequency Locked-Loop (SOGI-FLL)

- SOGI transfer functions analysis

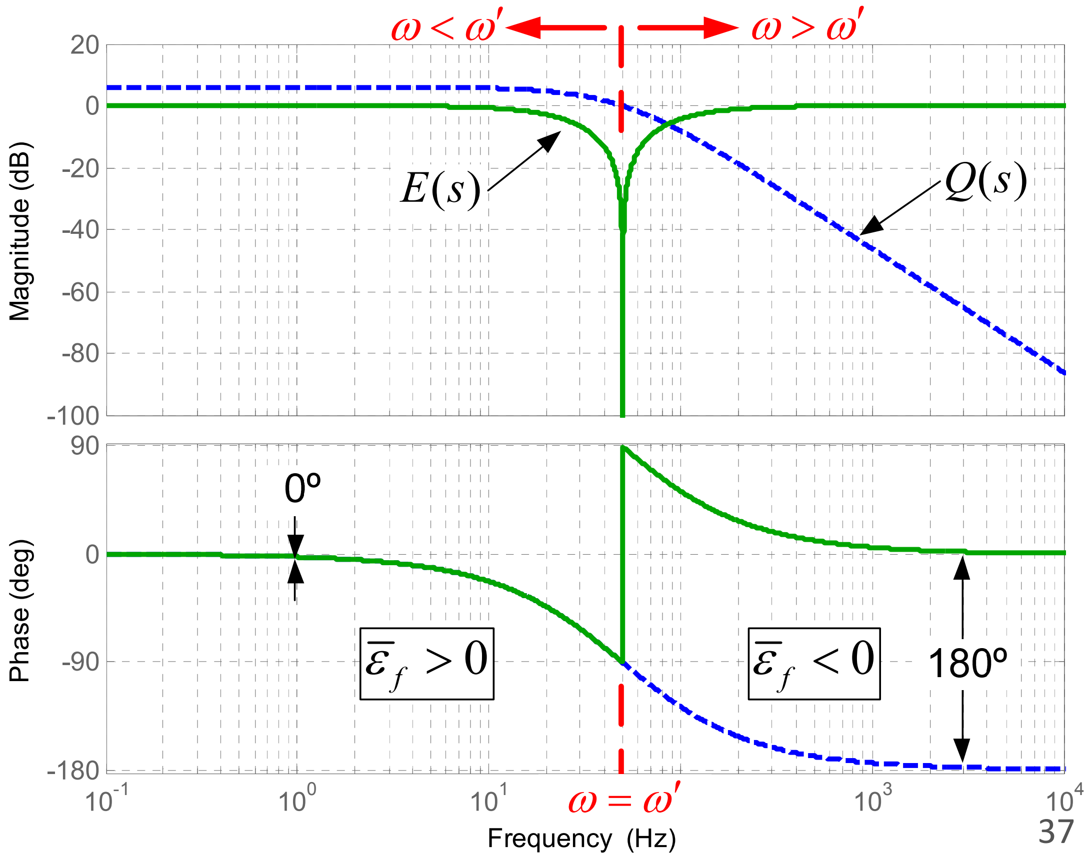
$$D(s) = \frac{v'}{v}(s) = \frac{k\omega' s}{s^2 + k\omega' s + \omega'^2} \qquad Q(s) = \frac{qv'}{v}(s) = \frac{k\omega'^2}{s^2 + k\omega' s + \omega'^2} \qquad E(s) = \frac{\varepsilon_v}{v}(s) = \frac{s^2 + \omega'^2}{s^2 + k\omega' s + \omega'^2}$$

- A frequency error variable ε_f is defined as:

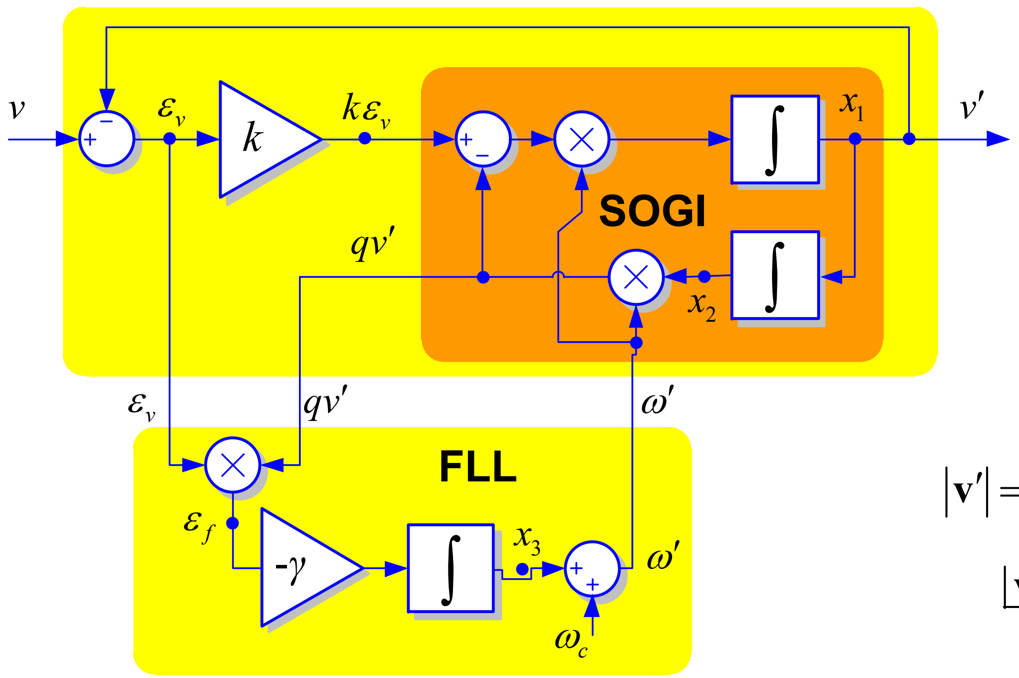
  $\varepsilon_f = \varepsilon_v \cdot qv'$

- The average value of ε_f will be positive when $\omega<\omega'$, zero when $\omega=\omega'$ and negative when $\omega>\omega'$

- An integral controller with a negative gain $-\gamma$, can be used to make zero the dc component of ε_f by shifting the AF resonance frequency ω' until matching the input frequency ω



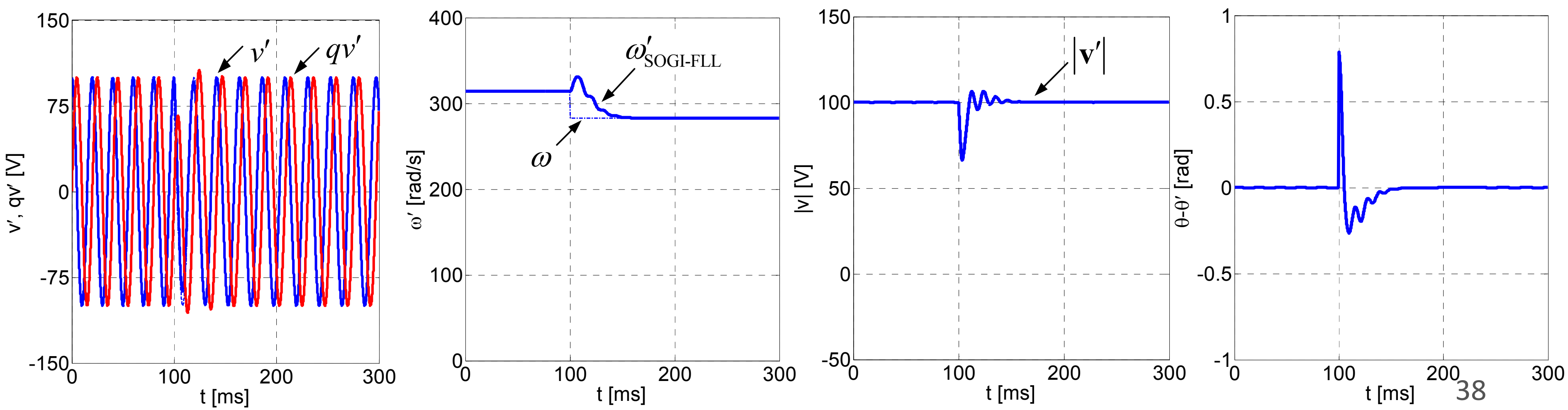
SOGI-Frequency Locked-Loop (SOGI-FLL)



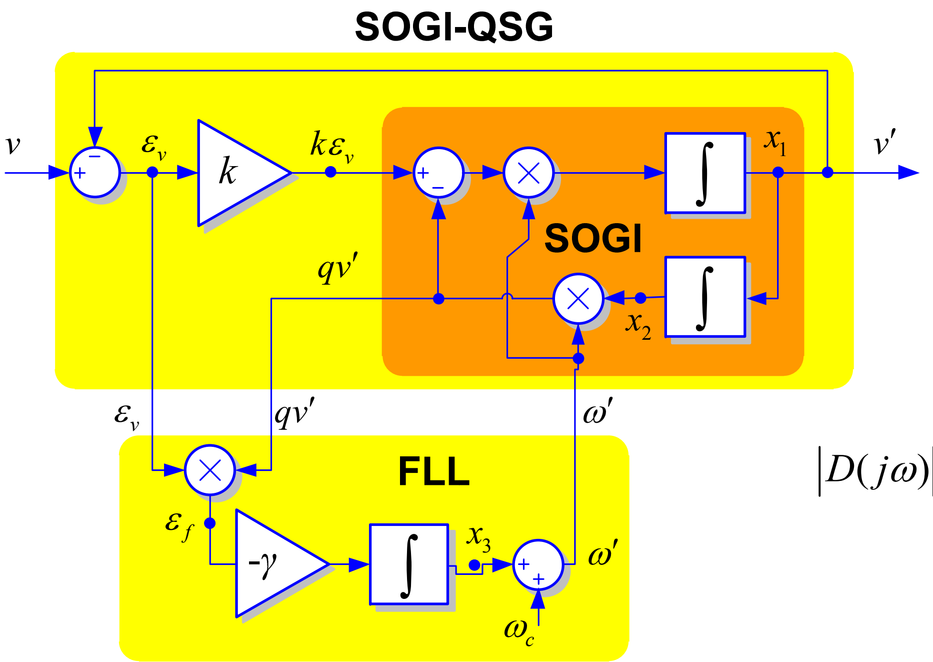
$$|\mathbf{v}'| = \sqrt{(v')^2 + (qv')^2}$$

$$\underline{|\mathbf{v}'} = \arctan \frac{qv'}{v'}$$

SOGI-FLL response:



Analysis of the SOGI-FLL



- Steady state response for a sinusoidal input
- $v = V\sin\left(\omega t+\phi\right)$ when the FLL is out of tuning

$$\bar{\mathbf{y}}' = \begin{bmatrix} v' \\ qv' \end{bmatrix} = V\left|D(j\omega)\right| \begin{bmatrix} \sin\left(\omega t+\phi+\angle D(j\omega)\right) \\ -\dfrac{\omega'}{\omega}\cos\left(\omega t+\phi+\angle D(j\omega)\right) \end{bmatrix}$$

$$\left|D(j\omega)\right| = \frac{k\omega\omega'}{\sqrt{\left(k\omega\omega'\right)^2+\left(\omega^2-\omega'^2\right)^2}} \qquad \angle D(j\omega) = \arctan\frac{\omega'^2-\omega^2}{k\omega\omega'}$$

State space equations

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \mathbf{A}\mathbf{x}+\mathbf{B}v = \begin{bmatrix} -k\omega' & -\omega'^2 \\ 1 & 0 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} k\omega' \\ 0 \end{bmatrix} v$$

$$\mathbf{y} = \begin{bmatrix} v' \\ qv' \end{bmatrix} = \mathbf{C}\mathbf{x} = \begin{bmatrix} 1 & 0 \\ 0 & \omega' \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$\dot{\omega}' = -\gamma x_2\omega'\left(v-x_1\right)$$

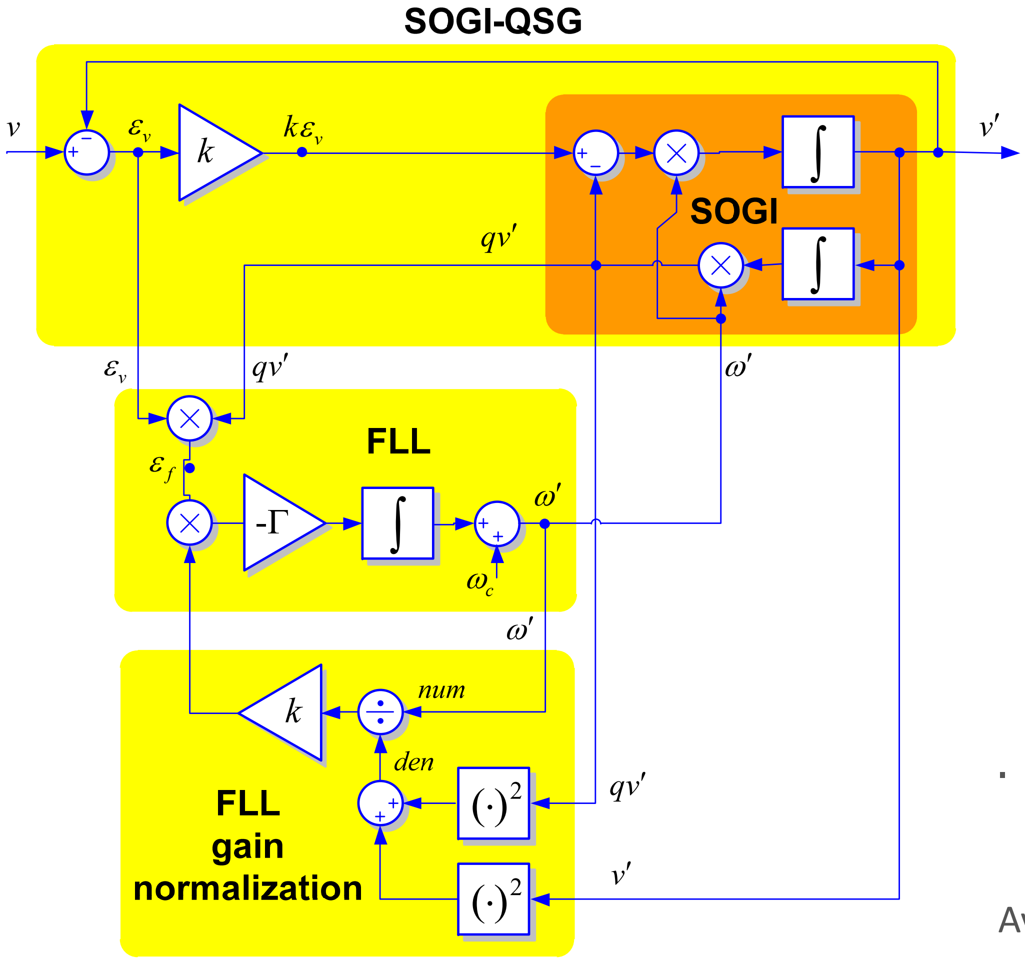
$$\dot{\bar{x}}_1 = -\omega^2\bar{x}_2$$

$$\bar{\varepsilon}_v = \left(v-\bar{x}_1\right) = \frac{1}{k\omega'}\left(\dot{\bar{x}}_1+\omega'^2\bar{x}_2\right)$$

$$\bar{\varepsilon}_f = \omega'\bar{x}_2\bar{\varepsilon}_v = \frac{\bar{x}_2^{\,2}}{k}\left(\omega'^2-\omega^2\right)$$

$$\dot{\omega}' = -\gamma\bar{\varepsilon}_f = \frac{\gamma}{k}\bar{x}_2^{\,2}\left(\omega'^2-\omega^2\right) \approx -2\frac{\gamma}{k}\bar{x}_2^2\left(\omega'-\omega\right)\omega'$$

Feedback based linearization of the FLL



$$\dot{\omega}' = -2\frac{\gamma}{k}\bar{x}_2^2\left(\omega'-\omega\right)\omega'$$

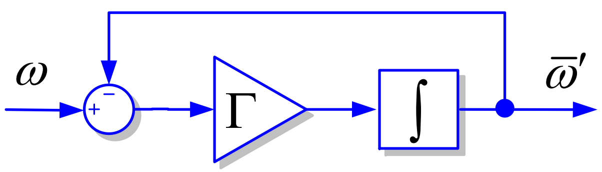
$$\bar{x}_2 = -\frac{V}{\omega}\left|D(j\omega)\right|\cos\left(\omega t+\phi+\angle D(j\omega)\right)$$

$$\bar{x}_2^2 = \frac{V^2}{2\omega^2}\left|D(j\omega)\right|^2\left[1+\cos\left(2\left(\omega t+\phi+\angle D(j\omega)\right)\right)\right]$$

$$\boxed{\dot{\bar{\omega}}' = -\frac{\gamma V^2}{k\omega'}\left(\bar{\omega}'-\omega\right)}$$

- Making $\gamma = \frac{k\omega'}{V^2}\Gamma$ we obtain $\dot{\bar{\omega}}' = -\Gamma\left(\bar{\omega}'-\omega\right)$

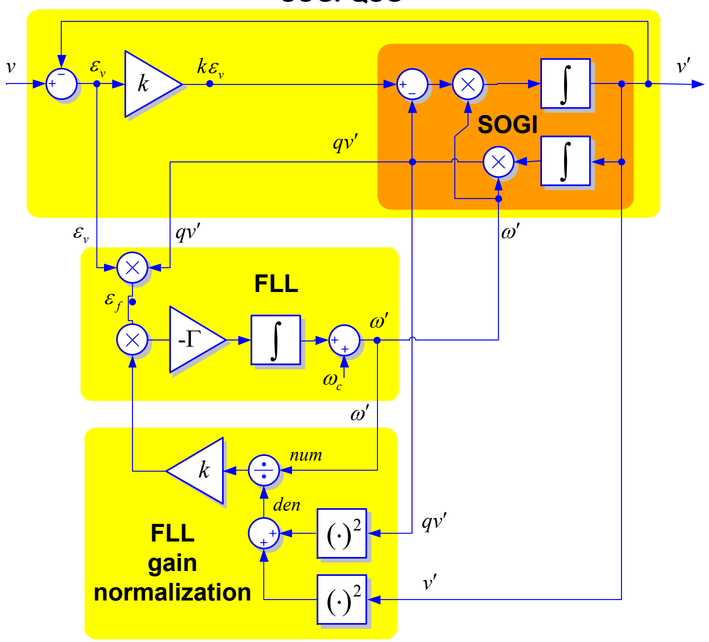
Averaged FLL dynamics



$$\boxed{\frac{\bar{\omega}'}{\omega} = \frac{\Gamma}{s+\Gamma}}$$

$$\boxed{t_{s(FLL)} \approx \frac{4.6}{\Gamma}}$$

SOGI-Frequency Locked-Loop (SOGI-FLL)



Tuning parameters:

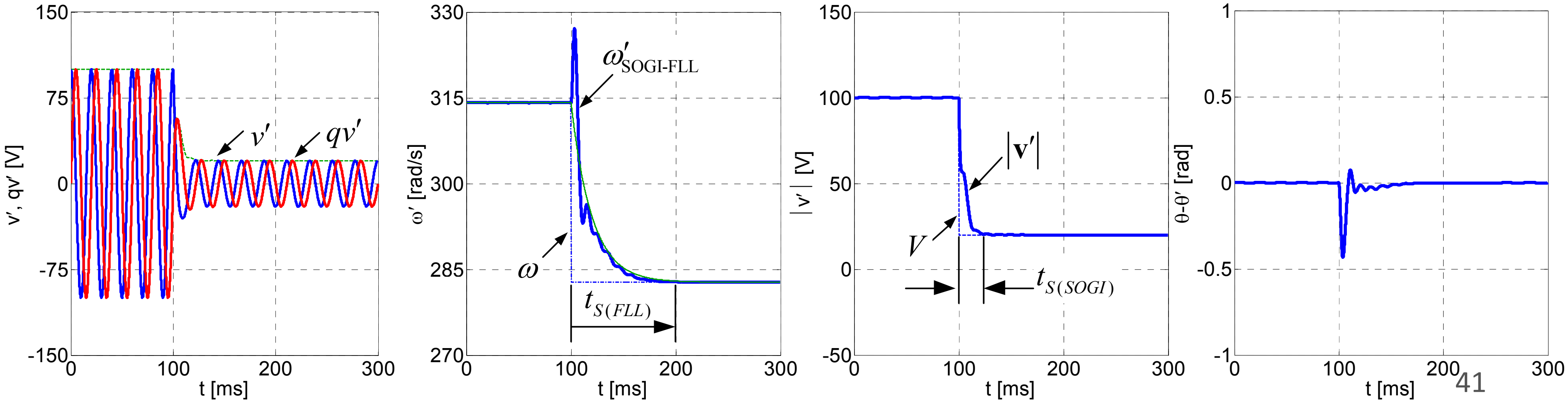
$$\boxed{t_{s(AF)} = \frac{9.2}{k\omega'} \; ; \; \xi = \frac{1}{\sqrt{2}}}$$

$$\boxed{t_{s(FLL)} \approx \frac{4.6}{\Gamma}}$$

Tuning condition:

$$t_{S(FLL)} \geq 2 \cdot t_{S(AF)}$$

Linearized SOGI-FLL response:



Conclusion

- Grid synchronization allows a proper instantaneous interaction between the power converter and the grid
- PLL is a very useful method that enable the grid inverters to:
  - Create a "clean" current reference synchronized with the grid voltage
  - Comply with the grid monitoring standards
- The PLL is able to track the frequency and phase of the input signal in a given settling time
- In-quadrature signal generation is an effective technique to improve the response of the single-phase PLL
- Transport delay, inverse Park transformation, or adaptive filters based on generalized integrators are some the methods used for quadrature signal generation
- The SOGI-FLL is a very effective method for single phase synchronization that makes the synchronization system frequency adaptive
- The analysis of the SOGI-FLL resulted in the linearization of the FLL response